INFORMATION TRUST, INFERENCE AND TRANSFER IN SOCIAL AND INFORMATION NETWORKS

BY

GUOJUN QI

DISSERTATION

Submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy in Electrical and Computer Engineering
in the Graduate College of the
University of Illinois at Urbana-Champaign, 2013

Urbana, Illinois

Doctoral Committee:

Professor Thomas Huang, Chair
Dr. Charu Aggarwal, IBM T.J. Watson Research Center
Professor Jiawei Han
Professor Mark Hasegawa-Johnson
Professor Zhi-Pei Liang

Abstract

In this thesis, our overarching goal is to aggregate crowdsourced information that is collected from computing systems based on social networks and represented in information networks. Due to the autonomous nature of such a social computing paradigm, the crowdsourced information is often subject to low quality, contributed by susceptible information sources without a reliant quality control scheme. Thus, to reveal the trustworthiness of the involved information sources, we aim to explore the social dependency behind the social networks where information contributors are prone to be influenced by each other. We explored the impact of such social dependency between sources on the information trust, aggregation and quality in social computing models. On the other hand, we will also investigate the structure underlying information shared by sources to reveal their trustworthiness. Our study will deepen our understanding of the patterns and behaviors of information sources and their reliability from both social and information aspects. Several closely related problems are investigated in this thesis: (1) the source trustworthiness, which aims to distinguish the untrustworthy sources from the trustworthy ones; (2) social signal processing, which aims to aggregate the multi-source contributed information to recover the true signals behind the problems such as the correct answers to a question and the true labels for an image; (3) the social dependency, which reveals the mutual influences among different sources; and (4) the nature of information structure, such as the information dependency underlying low-rank structure and visual similarities. Our goal is to propose a unified probabilistic model to explain the social and information phenomena behind these problems. In this thesis, we designed several algorithms which are tested in several real social and information network scenarios. Superior performances have been achieved compared with many existing state-of-the-art technologies in the areas.

To my parents, wife and son, for their love and support

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

With the proliferation of ubiquitous networking technologies, users can easily connect with each other on social and information networks where they can transmit, share and spread their observations and knowledge in a timely fashion. This opens unprecedented opportunity to integrate the crowdsourced information collected from distributed data sources in order to complete collaborative tasks based on the social and information network infrastructures. For example, users can turn to the online Q&A forums to seek the answers to their questions by consulting the peer users who have expertise knowledge in the relevant domains; the real-time update of social media platforms can be fused to facilitate early warning of and ensure timely response to the emergent social and natural events, outbreak of epidemics, and disastrous accidents. Compared with many traditional information systems based on the central databases and/or professional experts, the new information systems driven by the wisdom of the crowds are equipped with distributed data sources connected by the social and information networks. Thus they can provide decision makers and average users more affordable, up-to-date and comprehensive information services in a wide range of knowledge domains.

However, due to the autonomous nature of social and information networks, no central mechanism exists to control the quality of information shared by data sources. Accordingly, the crowdsourced information systems can be negatively impacted by the low-quality data sources in the social and information networks. Thus, to improve the decision-making quality of such information systems, we propose to evaluate the *information trust* shared by the crowds, and extract the high-quality information to build trustworthy distributed databases upon the social networks. The scientific goal of this research is to study *how the social connectivity and dependency affect the trustworthiness of distributed data sources*. To answer this question, we employ probabilistic models to explore the social groups which cluster the

dependent users who are prone to be mutually influenced. Such social groups are linked with the well-studied concept *community* in the social network analysis.

Besides the social connectivity structures, we also noted that the nature of data connections underlying the information networks also play a vital role in integrating the distributed databases. As a concrete example, we studied the *Multimedia Information Networks* which connect the context and the contextual information objects in a unified network structure. We explored the underlying connections between these information objects, and developed an *information inference* algorithm to infer a latent representation for the multimedia documents in the networks. This results in a robust classifier for multimedia content by eliminating the noisy links in the information networks.

Finally, we concentrate on heterogeneous networks with various types of objects. In many cases, crowdsourced information often consists of heterogeneous information objects. Integration of such heterogenous networks, especially discovering their cross-network structures, can provide useful clues to decision makers. For this purpose, we developed an *information transfer* algorithm to transfer the link structures between heterogeneous networks. We will show how the structural transfer can reveal the underlying connections between networks.

In brief, the overall goal of the study is to reveal the impact of social and information network structures on the information trust, inference and transfer in the context of collective intelligence contributed by the crowds. We wish to deepen our basic understanding of information sharing and spreading patterns from the network point of view. This might eventually lead to a robust collective computing model based on distributed crowdsourced information to enhance the intelligence power of decision makers and general users.

CHAPTER 2

INFORMATION TRUST

Collective intelligence, which aggregates the shared information from large crowds, is often negatively impacted by unreliable information sources with the low-quality data. This becomes a barrier to the effective use of collective intelligence in a variety of applications. In order to address this issue, we propose a probabilistic model to jointly assess the reliability of sources and find the true data. We observe that different sources are often not independent of each other. Instead, sources are prone to be mutually influenced, which makes them dependent when sharing information with each other. High dependency between sources makes collective intelligence vulnerable to the overuse of redundant (and possibly incorrect) information from the dependent sources. Thus, we reveal the latent group structure among dependent sources, and aggregate the information at the group level rather than from individual sources directly. This can prevent the collective intelligence from being inappropriately dominated by dependent sources. We will also explicitly reveal the reliability of groups, and minimize the negative impacts of unreliable groups. Experimental results on real-world data sets show the effectiveness of the proposed approach with respect to existing algorithms.

2.1 Introduction

Collective intelligence aggregates contributions from multiple sources in order to collect data for a variety of tasks. For example, voluntary participants collaborate with each other to create a fairly extensive set of entries in Wikipedia, or a crowd of paid persons may perform image and news article annotations in Amazon Mechanical Turk. These crowdsourced tasks usually involve multiple *objects*, such as Wikipedia entries and images to be anno-

tated. The participating sources collaborate to claim their own *observations*, such as facts and labels, on these objects. Our goal is to aggregate these collective observations to infer the *true values* (e.g., the true fact and image label) for the different objects [1], [2], [3].

We note that an important property of collective intelligence is that different sources are typically not independent of one another. For example, in the same social community, people often influence each other, where their judgments and opinions are not independent. In addition, task participants may obtain their data and knowledge from the same external information source, and their contributed information will be dependent. Thus, it may not be advisable to treat sources independently and directly aggregate the information from individual sources, when the aggregation process is clearly impacted by such dependencies. In this chapter, we will infer the source dependency by revealing latent group structures among involved sources. Dependent sources will be grouped, and their reliability is analyzed at the group level. The incorporation of such dependency analysis in group structures can reduce the risk of overusing the observations made by the dependent sources in the same group, especially when these observations are unreliable. This helps prevent dependent sources from inappropriately dominating collective intelligence especially when these sources are not reliable.

Moreover, we note that groups are not equally reliable, and they may provide incorrect observations which conflict with each other, either unintentionally or maliciously. Thus, it is important to reveal the reliability of each group, and minimize the negative impact of the unreliable groups. For this purpose, we study the *general* reliability of each group, as well as its *specific* reliability on each individual object. These two types of reliability are closely related. General reliability measures the overall performance of a group by aggregating each individual reliability over the entire set of objects. On the other hand, although each object-specific reliability is distinct, it can be better estimated with a prior that a *generally reliable* group is likely to be reliable on an individual object and vice versa. Such a prior can reduce the overfitting risk of estimating each object-specific reliability, especially considering that we need to determine the true value of each object at the same time [4], [5].

The remainder of this chapter is organized as follows. We review the related work in Section 2.2. Our problem and notations are formally defined

in Section 2.3. The probabilistic model for the problem is developed in Section 2.4, followed by a running example that illustrates the impact of group dependency on the model in Section 2.5. Section 2.6 presents the model inference and parameter estimation algorithms. Then Section 2.7 presents the application of the developed model to training classifiers from noisy crowdsourced data. We evaluate the model in Section 2.8 on real data sets. Section 2.9 and Section 2.10 give the details of model inference and parameter estimation, and Section 2.11 summarizes the chapter with the conclusion.

2.2 Related Work

Aggregating crowdsourced knowledge and information has attracted a lot of research efforts, and yields many insightful discoveries. For example, Yin et al. [6] proposed an iterative truth finder algorithm by simultaneously accessing the trustworthiness of each source and the correctness of claimed facts. Bachrach et al. [5] developed a probabilistic graphical model by jointly modeling the abilities of participants and the correct answers to questions in an aptitude testing setting. The work in [1] developed a latent truth model to infer the source quality and correct claims by modeling two types of false positive and false negative errors of each source. All of these algorithms estimate the performances of data sources and the impacts on the credibility of their claimed facts.

However, sources are not independent of each other in real world. Instead, their contributions are typically dependent. Yin et al. [6] noted this problem and used a dampening factor to compensate for excessively high confidence due to the copied content between sources. But this method did not explicitly model the dependency between sources, and how the dampening factor can reduce the dependency effect is not clear. On the other hand, the relation between the content claimed by sources, and a separate weighted voting algorithm by considering the copied content between each other have been studied in [7]. However, the accuracies are accessed *independently* on the source level, which can make the accuracy of a data source overestimated if many other dependent sources repeat the same false facts.

Moreover, existing models [7], [8], [9], [10] only consider the pairwise

relations between sources to their dependency, which completely ignores the higher-order dependency among sources. In contrast, we explicitly group the dependent sources to capture arbitrary orders of dependency among sources. We find that high-order dependency prevails in many real cases, and it is more effective to model them directly rather than decomposing them into separate pairwise relations. For example, sources which obtain the content from the same resource will be assigned to the same group to reflect the high order dependency among them. This yields a more compact representation to jointly assess the reliability of data sources and the correctness of the facts claimed. Moreover, we will see based on the group-level dependency, independent sources from different groups will play a more important role than dependent ones in the same group in inferring the true facts. This is a desired property which can properly aggregate collective knowledge in many real-world tasks.

Modeling the group dependency can be analogized to the community discovery in social networks. Community structure has been considered as a more effective data structure to capture the social relations among people than the links between pairs of persons [11]. With the similar spirit, the groups can also be more effective than pairwise dependency, and provide deeper insight into the property of high-order dependency among sources and how such a property affects the aggregation of collective knowledge. However, it is worth pointing out that the groups defined in our model differ from the communities [12] in social networks. Communities are usually defined as a set of people densely linked in social networks. However, two linked people may not necessarily be influenced by one another when they report the facts and knowledge. Two close friends can express different opinions and claim conflicting truths. Therefore, we will directly investigate the data contributed by sources to find the group structure characterizing their mutual dependency that directly affects the source reliability in our collective intelligence model.

Finally, our model is motivated to explore the objective facts and knowledge. This is in contrast to the inference of individual's preference, which aims to recommend products and services based on users' ratings and opinions [13]. Instead, in this chapter we aim at aggregation of collective knowledge to automatically extract the true facts, such as correct answers to questions and true categories for web pages, which do not depend on the variability of users' subjectivity.

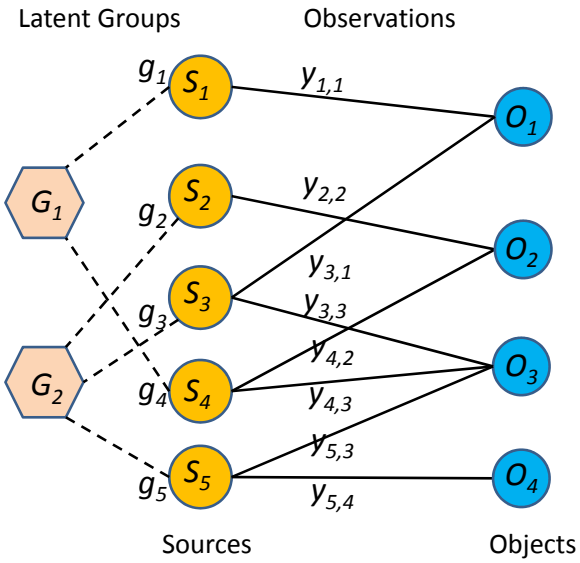


Figure 2.1: An example illustrating a set of five sources with their observations on four objects.

2.3 Problem Definitions

We formally define the following Multi-Source Sensing (MSS) model which abstracts the description of collective intelligence. Suppose that we have a set $\mathcal{S} := \{S_1, S_2, \cdots, S_N\}$ of N sources, and a set $\mathcal{O} := \{O_1, O_2, \cdots, O_M\}$ of M objects. Each object O_m takes a value t_m from a domain $\mathcal{X}_m$ which describes one of its attributes. Each source S_n in $\mathcal{S}$ reports its observation $y_{n,m} \in \mathcal{X}_m$ on an object O_m. Then the goal of the MSS model is to infer the true value t_m of each object O_m from the observations made by sources. We introduce some notations, which will be used consistently in this chapter. We will use n, m, l and k in the subscript to index sources, objects, groups and values in an object domain, respectively. The variables y, t, u and r denote the observations, true values, group reliability and object-specific reliability respectively.

In this chapter, we are particularly interested in a categorical domain $\mathcal{X}_m = \{1, \cdots, K_m\}$ with discrete values. For example, in many crowdsourcing applications, we focus on the (binary-valued) assertion correctness in a hypothesis test and (multi-valued) categories in a classification problem. However, the MSS model can be extended to the continuous domain with some effort by adopting the corresponding continuous distributions. Due to the space limitation, we leave this extension in the full version of this chapter.

Figure 2.1 illustrates an example, where five sources make their observations on four objects. An object can be an image or a biological molecule, and an annotator or a biochemical expert (as a source) may claim the category (as the value) for each object. Alternatively, an object can be a book, and a book seller web site (as a source) claims the identity of its authors (as the values). In a broader sense, objects are even not concrete objects. They can refer to any crowdsourced tasks, such as questions (e.g., "Is Peter a musician?") and assertions (e.g., "George Washington was born on February 22, 1732." and "an animal is present in an image,"), and the observations by sources are the answers to the questions, or binary-valued positive or negative claims on these assertions.

It is worth noting that each source does not need to claim the observations on all objects in $\mathcal{O}$. In many tasks, sources make claims only on small subsets of objects of interest. Thus, for notational convenience, we denote all claimed observations by $\mathbf{y}$ in bold, and use $I = \{(n,m)|\exists\ y_{n,m} \in \mathbf{y}\}$ to denote all the indices in $\mathbf{y}$. We use the notations $I_{n,\cdot} = \{m|\exists\ (n,m) \in I\}$ and $I_{\cdot,m} = \{n|\exists\ (n,m) \in I\}$ to denote the subset of indices that are consistent with the corresponding subscripts n and m.

Meanwhile, in order to model the dependency among sources, we assume that there is a set of latent groups $\{G_1, G_2, \cdots\}$, and each source S_n is assigned to one group G_{g_n} where $g_n \in \{1, 2, \cdots\}$ is a random variable indicating its membership. For example, as illustrated in Figure 2.1, the five sources are inherently drawn from two latent groups, where each source is linked to the corresponding group by dotted lines. Each latent group contains a set of sources which are influenced by each other and tend to make similar observations on objects. The unseen variables of group membership will be inferred mathematically from the underlying observations. Here, we do not assume any prior knowledge on the number of groups. The composition of these latent groups will be determined with the use of a Bayesian nonparametric approach by stick-breaking construction [14], which is presented in Section 2.4.

To minimize the negative impact of unreliable groups, we will explicitly model the group-level reliability. Specifically, for each group G_l, we define a group reliability score $u_l \in [0, 1]$ in unit interval. This value measures the general reliability of the group over the entire set of objects. A higher value of u_l indicates the greater reliability of the group.

Meanwhile, we also specify the reliability $r_{l,m} \in \{0, 1\}$ of each group G_l on each particular object O_m. When $r_{l,m} = 1$, group G_l will have reliable performance on O_m, and otherwise it will be unreliable. The reason that we distinguish between reliability u_l and object-specific reliability $r_{l,m}$ is as follows. While a generally reliable group with a larger value of u_l, provides very useful evidence about the members of the group on a generic basis, there are likely to be natural variations within the group itself. Thus, in our model, a group reliability u_l only measures how likely it will be reliable on the object set, and whether it will have a reliable performance on a particular object is given by $r_{l,m}$. In Section 2.4, we will clarify the relationship between general reliability u_l and object-specific reliability $r_{l,m}$.

2.4 Multi-Source Sensing Model

In this section, we present a generative process for the multi-source sensing problem. The output of this model will contain the following three aspects: (1) the group membership which describes the dependency between sources when claiming their observations on a set of objects; (2) the reliability u_l associated with each group and its specific reliability $r_{l,m}$ on each object; and (3) the true values t_m for each object. Our goal is to reveal the connections between these three aspects, especially how the collective observations made by sources can be explained by the latent groups and their reliability in a unified probabilistic framework.

First we define the following generative model for multi-source sensing (MSS) process that follows, the details of which will be explained shortly.

1. Draw $\boldsymbol{\lambda} \sim \text{GEM}(\kappa)$ (i.e., stick-breaking construction with concentration κ).

2. For each source S_n,

   2.1. Draw its group assignment $g_n|\boldsymbol{\lambda} \sim \text{Discrete}(\boldsymbol{\lambda})$.

3. For each object O_m,

   3.1. Draw its true value $t_m \sim \text{Uniform}(\mathcal{X}_m)$.

4. For each group G_l,

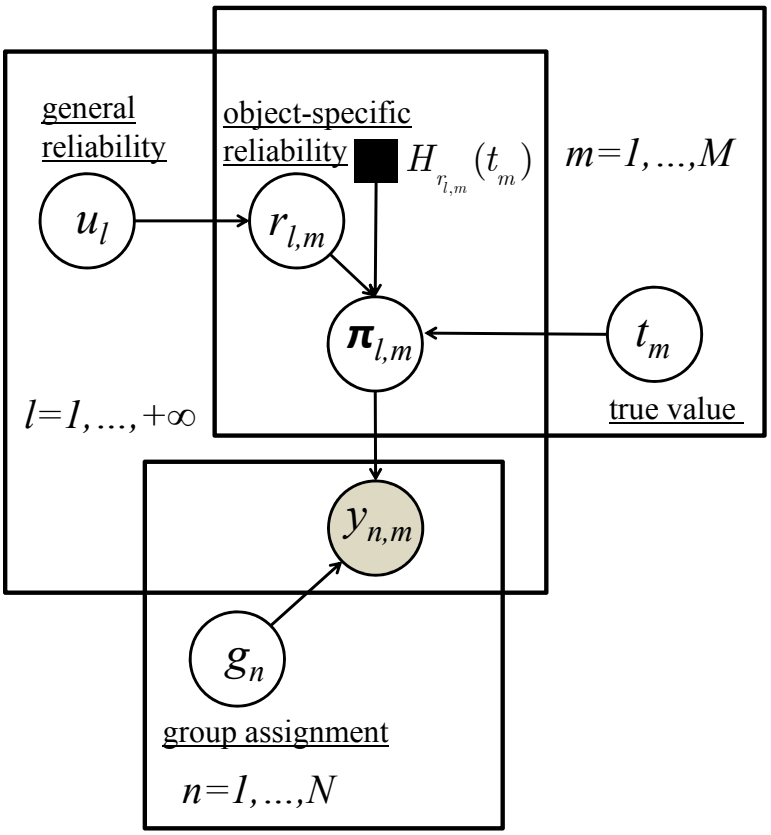


Figure 2.2: The graphical model for multi-source sensing. The three plates represent group reliability u_l with $l = 1, 2, \cdots$, the true values t_m for each object O_m with $m = 1, \cdots, M$, and the group assignment g_n of each source with $n = 1, \cdots, N$, respectively.

4.1. Draw its group reliability $u_l \sim \text{Beta}(b_1, b_0)$.

5. For each pair of group G_l and object O_m,

   5.1. Draw reliability indicator $r_{l,m} \sim \text{Bernoulli}(u_l)$, and

   5.2. Draw the observation model parameter

$$\boldsymbol{\pi}_{l,m} | r_{l,m}, t_m = z \sim H_{r_{l,m}}(t_m)$$

   for group G_l on object O_m.

6. For each $(n, m) \in I$,

   6.1. Draw observation $y_{n,m} | \boldsymbol{\pi}_{l,m}, g_n \sim F(\boldsymbol{\pi}_{g_n,m})$.

Here, $g_n | \boldsymbol{\lambda} \sim \text{Discrete}(\boldsymbol{\lambda})$ denotes a discrete distribution, which generates the value $g_n = l$ with probability λ_l; H and F are a pair of conjugate distributions which are determined by the type of data values on objects. For categorical values, these are Dirichlet and multinomial distributions, respectively. Figure 2.2 illustrates the generative process in a graphical representation. We will explain the details later.

In step 1, we adopt the stick-breaking construction GEM(κ) (named after Griffiths, Engen and McCloskey) with concentration parameter $\kappa \in \mathbb{R}^+$ to define the prior distribution of assigning each source S_n to a latent group G_{g_n} [14]. Specifically, in GEM(κ), a set of random variables $\boldsymbol{\rho} = \{\rho_1, \rho_2, \cdots\}$ are independently drawn from the beta distribution $\rho_i \sim \text{Beta}(1, \kappa)$. They define the mixing weights $\boldsymbol{\lambda}$ of the group membership component such that $p(g_n = l | \boldsymbol{\rho}) = \lambda_l = \rho_l \prod_{i=1}^{l-1} (1 - \rho_i)$. By the aforementioned stick-breaking process, we do not need the prior knowledge of the number of groups. This number will be determined by capturing the degree of dependency between sources.

Clearly, we can see that the parameter κ in the above GEM construction plays the vital role of determining a priori the degree of dependency between sources. According to the GEM construction, we can verify that the probability of two sources S_n and S_m being assigned to the same group is given

by the following:

$$\begin{aligned} P(g_n = g_m) &= \sum_{l=1}^{+\infty} \mathop{\mathbb{E}}_{\boldsymbol{\lambda}} P(g_n = l|\boldsymbol{\lambda}) P(g_m = l|\boldsymbol{\lambda}) \\ &= \sum_{l=1}^{+\infty} \mathop{\mathbb{E}}_{\lambda_l} \lambda_l^2 = \sum_{l=1}^{+\infty} \frac{2}{(1+\kappa)(2+\kappa)} \left(\frac{\kappa}{2+\kappa}\right)^{l-1} = \frac{1}{1+\kappa} \end{aligned} \tag{2.1}$$

It is evident that when κ is smaller, sources are more likely to be assigned to the same group where they are dependent and share the same observation model. This will yield a higher degree of dependency between sources. As κ increases, the probability that any two sources belong to the same group will decrease. In the extreme case, as $\kappa \to +\infty$, this probability approaches zero. In this case, all sources will be assigned to distinctive groups, yielding complete independence between sources. This shows that the model can flexibly capture the various degrees of dependency between sources by setting an appropriate value of κ.

In step 3, we adopt the uniform distribution as the prior on the true value t_m of each object over its domain $\mathcal{X}_m$. The uniform distribution sets an unbiased prior so that true values will be completely determined a posteriori given observations in the model inference. In Section 2.7, we will show how to set a more informative prior when more knowledge about objects is available.

In step 4, we define a Beta distribution $\text{Beta}(b_1, b_0)$ on the group reliability score u_l, where b_1 and b_0 are the soft counts which specify whether a group is reliable or not a priori, respectively. Then, in step 5.1, object-specific reliability $r_{l,m} \in \{0, 1\}$ is sampled from the Bernoulli distribution $\text{Bern}(u_l)$ to specify the group reliability on a particular object O_m. The higher the general reliability u_l, the more likely G_l is reliable on a particular object O_m with $r_{l,m}$ being sampled to be 1. This suggests that a generally more reliable group is more likely to be reliable on a particular object. In this sense, the general reliability serves as a prior to reduce the over-fitting risk of estimating object-specific reliability in the MSS model.

In step 5.2, the model parameter $\boldsymbol{\pi}_{l,m}$ for each group on a particular object is drawn from the conjugate prior $H_{r_{l,m}}(t_m)$, which depends on the true value t_m and the object-specific group reliability $r_{l,m}$. Then, given the group membership g_n, each source S_n generates its observation $y_{n,m}$ according to the corresponding group observation model $F(\boldsymbol{\pi}_{g_n,m})$ in step 6. In the next subsection, we will detail the specification of $H_{r_{l,m}}(t_m)$ and $F(\boldsymbol{\pi}_{l,m})$ in

categorical domain.

2.4.1 Group Observation Models

In this subsection, we discuss the specification of group observation distribution $F(\boldsymbol{\pi}_{l,m})$ and its conjugate distribution $H_{r_{l,m}}(t_m)$ for categorical values on each object. Here the group observation model on each object depends on two factors: (1) the specific reliability $r_{l,m}$ on this object, which aims to reveal the differences between reliable and unreliable observations on an object, and (2) the true value t_m for the object.

It is worth noting that although we distinguish each group observation into reliable and unreliable cases in this subsection, it does not mean that two groups are enough to capture the source dependency. These two cases are used to model the performance at the *object* level. However, given more objects, there are many possible combinations of these two cases on different objects. This is why we need more groups to capture the source dependency based on their observations on different objects. In the following, we will discuss the group observation models on each object.

In categorical domains, for each group, we choose the multinomial distribution as its observation model to generate each observation $y_{n,m}$ for its member sources on each object O_m. Thus, step 6 in the generative process of the MSS model becomes the following:

$$y_{n,m}|\boldsymbol{\pi}_{l,m}, g_n \sim F(\boldsymbol{\pi}_{g_n,m}) \triangleq \text{Multinomial}(\boldsymbol{\pi}_{g_n,m})$$

where $\boldsymbol{\pi}_{l,m}$ is the parameter of multinomial distribution for group G_l on object O_m. Here, all member sources in the same group share the same observation model to capture their dependency.

The model parameter $\boldsymbol{\pi}_{l,m}$ is generated by the following:

$$\begin{aligned}&\boldsymbol{\pi}_{l,m}|r_{l,m}, t_m = z \sim H_{r_{l,m}}(t_m)\\ &\triangleq \text{Dir}(\underbrace{\theta^{(r_{l,m})}, \cdots,}_{z-1} \underset{\substack{\downarrow\\ z^{\text{th}}\ \text{entry}}}{\eta^{(r_{l,m})}}, \cdots, \theta^{(r_{l,m})})\end{aligned}$$

where Dir denotes Dirichlet distribution, and $\theta^{(r_{l,m})}$ and $\eta^{(r_{l,m})}$ are its soft counts for sampling the false and true values under different settings of $r_{l,m}$.

If group G_l has reliable observations for object O_m (i.e., $r_{l,m} = 1$), it should be more likely to sample the true value $t_m = z$ as its observation than sampling any other false value. Thus, we should set a larger value for $\eta^{(r_{l,m})}$ than for $\theta^{(r_{l,m})}$.

On the other hand, if group G_l has unreliable observations for object O_m, i.e., $r_{l,m} = 0$, it should *not* be more likely to claim the true value for the object than claiming the false values. Therefore, the group observation model should have $\eta^{(0)}$ no larger than $\theta^{(0)}$, i.e., $\eta^{(0)} \leq \theta^{(0)}$. Specifically, the mathematical model can distinguish between *uninformative* and *malicious* observations on the target object:

I. **Uninformative observation:** When $\eta^{(0)} = \theta^{(0)}$, sources in group G_l make uninformative observations on object O_m, since false values are equally likely to be claimed as the true value. This can be caused when these sources either carelessly claim their observations at random, or lack the knowledge about the target object.

II. **Malicious observation:** When $\eta^{(0)} < \theta^{(0)}$, it suggests that the group G_l contains malicious sources which tend to claim false values for object O_m. Compared with uninformative observations, these malicious observations can even provide us with some information about the target object by interpreting the observations in a reverse manner. Actually, with $\theta^{(0)} > \eta^{(0)}$, the model gives the unclaimed observation larger weight to be evaluated as the true value.

In summary, depending on $r_{l,m}$, the sources in group G_l make either reliable (when $r_{l,m} = 1$) or unreliable (when $r_{l,m} = 0$) observations on a particular object O_m. Accordingly, the corresponding parameters $\eta^{(r_{l,m})}$ and $\theta^{(r_{l,m})}$ are constrained in different ways. When $r_{l,m} = 1$, we impose a strict inequality $\eta^{(1)} > \theta^{(1)}$ to enforce that group G_l is more likely to claim the true value. On the contrary, when $r_{l,m} = 0$, we have $\theta^{(0)} \geq \eta^{(0)}$, representing that G_l will be unreliable in terms of claiming the true value for O_m. In Section 2.10, we will see how these parameters can be estimated by maximizing the observation likelihood of the MSS model subject to these constraints.

By putting together these different pieces, the MSS defines a complete

distribution

$$p(\mathbf{y}, \mathbf{g}, \mathbf{r}, \mathbf{u}, \mathbf{t}, \boldsymbol{\pi}|\Theta) = \prod_{m=1}^{M} p(t_m) \prod_{l=1,m=1}^{L,M} p(u_l|b_1, b_0)p(r_{l,m}|u_l)$$
$$\times p(\pi_{l,m}|r_{l,m}, t_m, \eta^{(r_{l,m})}, \theta^{(r_{l,m})})$$
$$\times \prod_{n=1}^{N} p(g_n|\kappa) \prod_{(n,m)\in I} p(y_{n,m}|g_n, \pi_{g_n,m})$$

over $\mathbf{g} = \{g_n\}$, $\mathbf{r} = \{r_{l,m}\}$, $\mathbf{u} = \{u_l\}$, $\mathbf{t} = \{t_m\}$, $\boldsymbol{\pi} = \{\boldsymbol{\pi}_{l,m}\}$ and the source observations $\mathbf{y}$ with model parameters $\Theta = \{\eta^{(0)}, \theta^{(0)}, \eta^{(1)}, \theta^{(1)}, b_1, b_0, \kappa\}$. In Section 2.10, we will present how to infer (1) the true values t_m for each object, (2) group assignment g_n of each source, and (3) the general reliability u_l of each group and its specific reliability $r_{l,m}$ on each object from the MSS model a posteriori given the observations $\mathbf{y}$.

2.4.2 Multiple Attributes

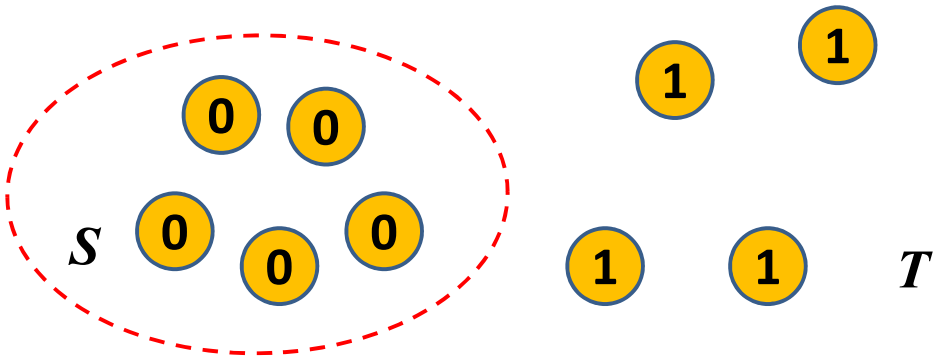
In some cases, an object might have multiple attributes. There are many such examples as follows.

- A person can have many attributes. For example, a person has a hobby of playing piano and takes "software engineer" as a vocation. We can consider hobby and vocation as two attributes for each person, and define their values on two different domain sets such as {playing piano, hiking, swimming, traveling $\cdots$} and {software engineer, stock trader, university faculty, $\cdots$} in MSS model, respectively.

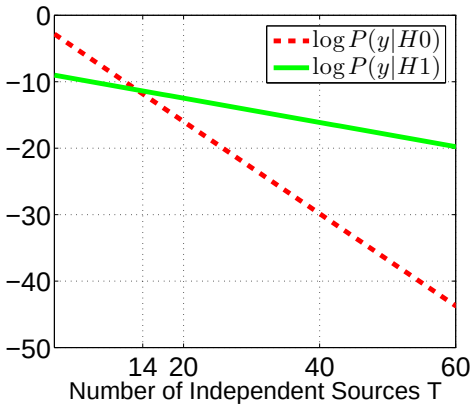
- An image can be labeled as "tiger" as well as "forest". We can consider the presence of these two nonexclusive labels as two different attributes, and their values are Boolean {Present, Not Present} for an image. In this way, we can allow an image has multiple labels simultaneously.

- A movie can have multiple actors/actresses. We can treat each actor/actress as an attribute, and use a binary value {1,0} to denote whether an actor/actress participates in a particular movie or not.

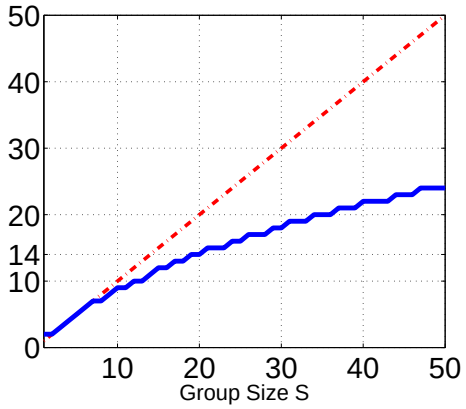
We can see in these examples, our MSS model is more flexible to handle multiple attributes associated with each object. Moreover, we note that different attributes often correlate with each other. For example, image labels



(a) An running example



(b) Likelihoods of two hypotheses



(c) Minimal number of independent sources to overturn the claims by S dependent sources (solid blue curve)

Figure 2.3: (a) A running example with S dependent sources in the same group and T independent sources. (b) Comparison of the likelihoods of two hypotheses (in Y-axis) versus varying number T of independent sources (in X-axis). The number of dependent sources in the group is fixed to $S = 20$. (c) The minimal number of independent sources (in Y-axis) to overturn the claims made by varying number of dependent sources (in X-axis). The results are obtained with $\eta^{(1)} = 10, \theta^{(1)} = 5$, and $\eta^{(0)} = \theta^{(0)} = 10$.

"tiger" and "forest" often co-occur in an image, and some actors/actresses may tend to co-star a movie. Exploring these attributes together can improve the accuracy of inferring their true values.

2.5 Dependence vs. Independence: A Running Example

In this section, we show a running example that demonstrates how group reliability structure captures the dependency between sources when it infers the true value for an object. In Figure 2.3(a), we show a group of S sources and T independent sources. We consider an ideal case where the S sources in the group make a unanimous claim of the value 0 for an object, while the T independent sources unanimously claims the opposite value 1 for the same object. While the dependent sources in the group and the independent sources claim the different values in this example, we can investigate different values of information contributed by these sources. Especially, we wonder whether independent sources play more important roles than dependent ones in finding the true value for each object in the MSS model.

For this purpose, we test the following two hypotheses:

- $H0$: The true value for the object is 0, versus
- $H1$: The true value for the object is 1.

To decide which hypothesis is true, we compare the observation likelihoods given these two hypotheses in the MSS model. Figure 2.3(b) compares the two likelihoods with varying number T of independent sources. The number of dependent sources is fixed to $S = 20$. We can see with more than $T = 14$ independent sources, $H1$ has a larger likelihood than $H0$. In this case, the claims made by independent sources become more credible than those made by dependent sources. This example shows fewer independent sources can overturn the claim made by more dependent sources. This suggests that each dependent source contains less information about the true claim as compared with each independent source.

To make this point more clear, Figure 2.3(c) illustrates the minimum number of independent sources to ensure $p(\mathbf{y}|H1) > p(\mathbf{y}|H0)$ under a varying

number of dependent sources S in the group. We can see that usually fewer independent sources are needed to have its claim accepted compared with the same number of dependent sources. This shows that independent sources are more valuable than dependent sources in determining the true value for each object. This is a desired property in our model, since we would like to de-emphasize the excessive impacts of dependent sources in a group.

Of courses, in the real world, sources may not be ideally split into dependent ones in a group, and completely independent ones. The independent sources may not make unanimous claims as in this case. However, this intuitive running example explains how the dependency encoded in group structure will affect the inference of true value on an object, and illustrates the independent claims are generally more valuable than dependent claims in the MSS model.

2.6 Model Inference and Parameter Estimation

In this section, we present the inference and learning processes. We wish to infer the tractable posterior $p(\mathbf{g}, \mathbf{r}, \mathbf{u}, \mathbf{t}, \boldsymbol{\pi}|\mathbf{y})$ with a parametric family of variational distributions in the factorized form:

$$q(\mathbf{g}, \mathbf{r}, \mathbf{u}, \mathbf{t}, \boldsymbol{\pi}) = \prod_n q(g_n|\boldsymbol{\varphi}_n) \prod_{l,m} q(r_{l,m}|\boldsymbol{\tau}_{l,m})$$
$$\prod_l q(u_l|\boldsymbol{\beta}_l) \prod_m q(t_m|\boldsymbol{\nu_m}) \prod_{l,m} q(\boldsymbol{\pi}_{l,m}|\boldsymbol{\alpha}_{l,m})$$

with parameters $\boldsymbol{\varphi}_n$, $\boldsymbol{\tau}_{l,m}$, $\boldsymbol{\beta}_l$, $\boldsymbol{\nu}_m$ and $\boldsymbol{\alpha}_{l,m}$ for these factors. The distribution and the parameter for each factor can be determined by the variational approach [15]. Specifically, we aim to maximize the lower bound of the log likelihood $\log p(\mathbf{y})$, i.e.,

$$\log p(\mathbf{y}) \geq \mathop{\mathbb{E}}_q \log p(\mathbf{g}, \mathbf{r}, \mathbf{u}, \mathbf{t}, \boldsymbol{\pi}, \mathbf{y}) - \mathop{\mathbb{E}}_q(\log q(\mathbf{g}, \mathbf{r}, \mathbf{u}, \mathbf{t}, \boldsymbol{\pi})) \triangleq \mathcal{L}(q)$$

This can obtain the optimal factorized distribution. The lower bound can be maximized over one factor while the others are fixed. This is an approach which is similar to coordinate descent. In each iteration, all the factors are updated sequentially over steps by finding the fixed-point solutions until convergence. The details of these updating steps are provided in Section 2.9.

We analyze the computational complexity in one loop of updating all factors. Suppose that we are given N sources, M objects, and obtain L groups by the stick-breaking construction. We also denote by $K_{\max}$ the maximum size of the domain sets among all objects. Then by investigating the updating steps in Section 2.9, we can find that the computational complexity is $O(NMLK_{\max})$ for one loop.

On the other hand, the model parameters Θ can be estimated by maximizing the observation likelihood. This can be done by the EM algorithm:
E-Step: Given the current parameters in Θ, apply variational inference to obtain the factorization q and their variational parameters;
M-Step: Given the factorization q, maximize the lower bound $\mathcal{L}(q)$ of the log-likelihood and obtain a new model parameter Θ. (Details of this maximization step are given in Section 2.10.)

These two steps are iterated until convergence. We obtain the variational approximation and the maximum likelihood parameter estimation results simultaneously.

2.7 Classification Problems

We are often particularly interested in the classification problem where each object takes a class as its value from a K-class domain $\mathcal{X} = \{1, 2, \cdots, K\}$. Moreover, we might be able to access the feature representations for the objects in $\mathcal{O}$. For example, if the objects are genetic sequences or text documents, we can extract their feature descriptors to describe the genetic structure and document content. Therefore, we wish to impose a more informative prior that aggregates these features into the prior distribution. For this purpose, given a feature vector $\mathbf{x}_m$ for an object, the prior on t_m becomes a conditional distribution on $\mathbf{x}_m$. For greater modeling flexibility, we choose a distribution for this prior. For example, we can choose an exponential distribution $p(t_m|\mathbf{x}_m, W)$:

$$\mathrm{Exp}(W) := p(t_m|\mathbf{x}_m, W) = \frac{1}{Z} \exp\left\{ \sum_{k=1}^{K} \delta\left[\!\left[t_m = k \right]\!\right] \langle \mathbf{w}_k, \mathbf{x} \rangle \right\} \tag{2.2}$$

where each coefficient vector is taken from the parameters $W = \{\mathbf{w}_k | k \in \mathcal{X}\}$, $\langle \mathbf{w}_k, \mathbf{x} \rangle$ denotes the inner product between two vectors, and Z is the normalization factor to ensure that the exponential distribution in Eq. (2.2) integrates to unit value.

Accordingly, the model inference in step 4 in Section 2.9 should be changed. Each updated factor $q(t_m)$ in model inference becomes an exponential distribution:

$$q(t_m|\boldsymbol{\nu}_m) := \exp\{\sum_{k=1}^{K} \delta\left[\!\left[t_m = k\right]\!\right] \nu_{m;k}\} \tag{2.3}$$

with the parameter $\boldsymbol{\nu}_m$ defined as follows:

$$\begin{aligned}\nu_{m;k} = \langle \mathbf{w}_k, \mathbf{x} \rangle + \sum_{l} \sum_{r_l} q(r_l)\{(\eta^{(r_l)} - 1) \\ \times \mathop{\mathbb{E}}_{q(\boldsymbol{\pi}_{l,m})} \ln \pi_{l,m;k} + \sum_{k' \neq k} (\theta^{(r_l)} - 1) \mathop{\mathbb{E}}_{q(\boldsymbol{\pi}_{l,m})} \ln \pi_{l,m;k'}\}\end{aligned}$$

The other updating steps for the model inference in Section 2.9 stay the same.

Besides the inference, we need to learn the parameter W in $p(t_m|\mathbf{x}_m, W)$. Here, we adopt the variational EM (Expectation-Maximization) algorithm. In each iteration, the E-step (expectation) involves computing the tractable posterior distributions as in the inference step. Then, the maximization step will update W by maximizing the expected log-likelihood over q as follows:

$$\max_{W} \sum_{m=1}^{M} \mathbb{E}_{q(t_m|\boldsymbol{\nu}_m)} \log p(t_m|\mathbf{x_m}, W) \tag{2.4}$$

We can adopt any off-the-shelf optimization algorithms to solve the above problem.

The learned parameterized model $p(t_m|\mathbf{x}, W)$, as a byproduct, is a classifier conditional on the input feature vector $\mathbf{x}$. This provides us with a way to train a robust classification model with the noisy crowdsourced labels, compared with typical classifiers trained with the clean labels. On the other hand, the learned classifier enhances the MSS model by providing a more discriminative prior for the labeling information on objects through their feature representations. This regularizes the true classes of objects in the feature space, especially when the classes claimed by different sources on an

object are too scarce or too inconsistent to make robust estimation of the true classes. In this case, the imposed prior plays a nontrivial role in determining the true class of the object.

2.8 Experimental Results

In this section, we compare our approach with other existing algorithms and demonstrate its effectiveness for inferring source reliability together with the true values of objects. The comparison is performed on a book author data set from online book stores, and a user tagging data set from the online image sharing web site `Flickr.com`.

2.8.1 Online Book Store Data Set

The first data set is the book author data set prepared in [6]. The data set is obtained by crawling $1,263$ computer science books on AbeBooks.com. For each book, AbeBooks.com returns the book information extracted from a set of online book stores. This data set contains a total of 877 book stores (sources), and $24,364$ listings of books (objects) and their author lists (object values) reported by these book stores. Note that each book has a different categorical domain that contains all the authors claimed by sources. Our goal is to predict the true authors for each book.

Author names are normalized by preserving the first and last names, and ignoring the middle name of each author. For evaluation purposes, the authors of 100 books are manually collected from scanned book covers [6]. We compare the returned results of each model with the ground truth author lists on this test set and report the accuracy.

We compare the proposed algorithm MSS with the following baselines: (1) the naive voting algorithm which counts the top voted author list for each book as the truth; (2) TruthFinder [6]; (3) Accu [7] which considers the dependency between sources; (4) 2-Estimates as described in [3] with the highest accuracy among all the models in [3].

Table 2.1 compares the results of the different algorithms on the book author data set in terms of the accuracy. The MSS model achieves the best accuracy among all the compared models. We note that the proposed MSS

Table 2.1: Comparison of different algorithms on book author and Flickr data set. On the book author data set, the algorithms are compared by their accuracies. On the Flickr data set, the algorithms are compared by their average precisions and recalls on 12 tags.

Model	book author data set	Flickr data set	
	accuracy	precision	recall
Voting[7]	0.71	0.8499	0.8511
2-Estimates[3]	0.73	0.8545	0.8602
TruthFinder[16]	0.83	0.8637	0.8649
Accu[7]	0.87	0.8731	0.8743
MSS	**0.95**	**0.9176**	**0.9212**

model is an unsupervised algorithm which does not involve any training data. In other words, we do not use any true values in the MSS algorithm in order to produce the reliability ranking as well as other true values. Even compared with the accuracy of 0.91 of the Semi-Supervised Truth Finder (SSTF) [16] using extra training data with known true values on some objects, the MSS model still achieves the highest accuracy of 0.95.

Figure 2.4(a) illustrates the scatter plot between the predicted reliability u_l for each group and its test accuracy. From this figure, it is evident that the group reliability obtained from the MSS model is a good predictor of the true accuracy for each group.

Meanwhile, we also report three example groups in Table 2.2. It is evident that within each group, the member sources have much consistent reliability as they make dependent claims. Therefore, by accurately predicting reliability level of groups, the proposed MSS model can appropriately aggregate the contributions from different groups and gain the competitive accuracy.

Moreover, to compare the reliability of sources, we can define the reliability of each source S_n by the expected reliability score of its assigned groups as follows:

$$\text{Reliability}(S_n) = \sum_l q(g_n = l) \mathop{\mathbb{E}}_{q(u_l|\boldsymbol{\beta}_l)}[u_l]$$

where

$$\mathop{\mathbb{E}}_{q(u_l|\boldsymbol{\beta}_l)}[u_l] = \frac{\beta_{l,1}}{\beta_{l,1} + \beta_{l,2}}$$

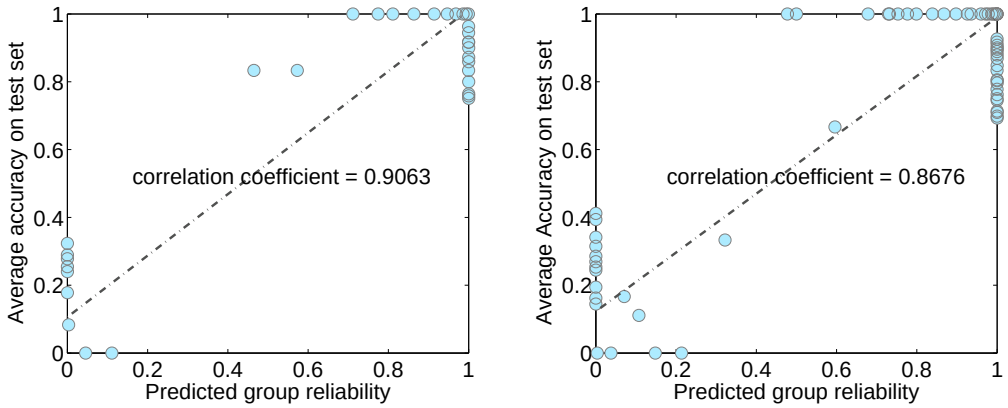
Then, sources can be ranked based on such source reliability. In Table 2.3, we rank the top-10 and bottom-10 book stores in this way. In order to show the

Table 2.2: Three example groups among all 33 groups discovered by the MSS model on book author data set. The number in parentheses after the name of each bookstore is its accuracy on the test set.

Group I	Group II	Group III
FREE U.S. AIR SHIPPING (0.3750)	The Book Depository (0.3043)	DVD Legacy (0.5833)
TheBookCom (0.3556)	textbookxdotcom (0.4444)	Englishbookservice.com (0.5500)
Browns Books (0.3438)	Caiman (0.3855)	Henry's Biz Books (0.6000)
Mellon's Books (0.4000)	Bobs Books (0.4615)	Blackwell Online (0.6579)
	Books Down Under (0.4750)	Morgenstundt Buch & Kunst (0.6207)
	Limelight Bookshop (0.3896)	
	Powell's Books (0.3810)	

Table 2.3: Top-10 and bottom-10 book stores ranked by their posterior probability of belonging to a reliable group. We also report the accuracy of these bookstores on the test set.

top-10 bookstores	accuracy	bottom-10 bookstores	accuracy
International Books	1	textbooksNow	0.0476
happybook	1	Gunter Koppon	0.225
eCampus.com	0.9375	www.textbooksrus.com	0.3333
COBU GmbH & Co. KG	0.875	Gunars Store	0.2308
HTBOOK	1	Indoo.com	0.3846
AlphaCraze.com	0.8462	Bobs Books	0.4615
Cobain LLC	1	OPOE-ABE Books	0
Book Lovers USA	0.8667	The Book Depository	0.3043
Versandantiquariat Robert A. Mueller	0.8158	Limelight Bookshop	0.3896
THESAINTBOOKSTORE	0.8214	textbookxdotcom	0.4444



(a) Book author data set (b) Flickr data set

Figure 2.4: Scatter plots on two data sets. The horizontal axis represents the predicted group reliability by u_l and the vertical axis represents the average accuracy of the member sources on the test set. The slope of each red line in the scatter is the correlation coefficient which shows the statistical correlation between u_l and the average accuracy.

extent to which this ranking list is consistent with the real source reliability, we provide the accuracy of these bookstores on test data sets. Note that each individual bookstore may only claim on a subset of books in the test set, and the accuracy is computed based on the claimed books. From Table 2.3, we can see that the obtained rank of data sources is consistent with the rank of their accuracies on the test set. On the contrary, the accuracy of the bottom-10 bookstores is much worse compared to that of the top-10 book stores on the test set. This also partially explains the better performance of the MSS model.

Since κ influences the dependency modeling between sources, we study the sensitivity of the model accuracy versus κ in Figure 2.5. We know that when $\kappa = 0$, all sources are completely dependent, and assigned to the same group. At this point, the model has a much lower accuracy, since all sources are tied to the same level of reliability within a single group. As κ increases, the accuracy achieves the peak at $\kappa = 5.0$. After that point, it deteriorates as the model gradually stops capturing the source dependency with increased κ. This demonstrates the importance of modeling the source dependency, and the capability of the MSS model in capturing such dependencies with κ.

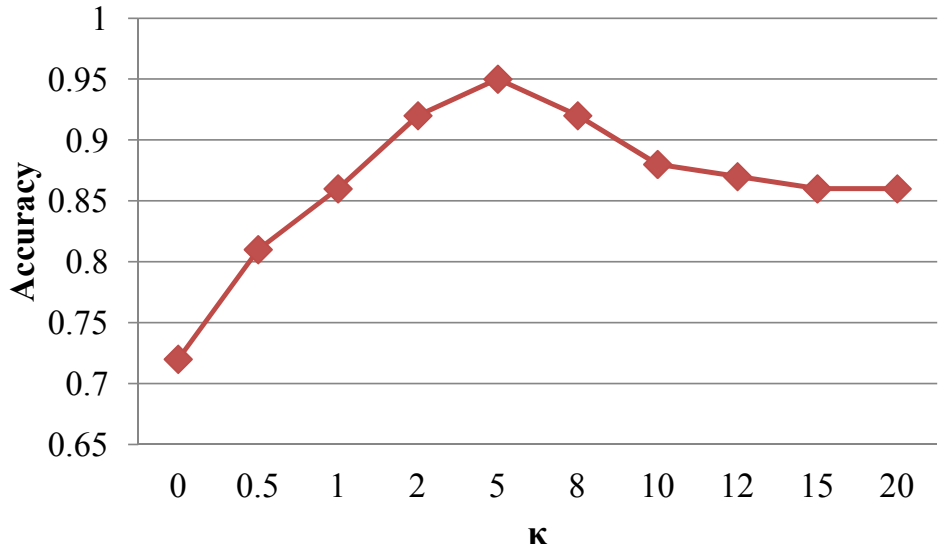


Figure 2.5: This figure illustrates the parametric sensitivity, i.e., model accuracy versus different κ on book author data set.

2.8.2 Flickr Image Tagging Data Set

We also evaluate the algorithm on a user tagging data set from an online image sharing web site Flickr.com. This data set contains $13,528$ users (data sources) who annotate $36,280$ images (data objects) with their own tags. We consider 12 tags – "balloon," "bird," "box," "car," "cat," "child," "dog," "flower," "snow leopard," "waterfall," "guitar," and "pumpkin" for evaluation purposes. Each tag is associated with a binary value 1/0 to represent its presence or not in an image. This forms a multi-attribute model with these 12 tags to find whether they are present on each image. Different from the book author data set, we apply the extended classification model in Section 2.7, where the visual content of each image is represented by a $8,000$ dimensional hierarchical Gaussian [17] feature vector.

Figure 2.6 illustrates some image examples in this data set and the tags annotated by users. It is evident that some images are wrongly tagged by users. The MSS model aims to correct these errors and yield accurate annotations on these images. To test accuracy, we manually annotate these 12 tags on a subset of $1,816$ images.

We follow the same experimental setup as on the book author data set. For the sake of fair comparison, we adopt the variants in [18] to incorporate visual features to enhance the original algorithms for comparison by inferring the

(a) balloon (b) snow leopard

(c) guitar (d) pumpkin

Figure 2.6: Examples of image and the associated user tags in Flickr data set. In each subfigure the left image is correctly tagged by users, while the right one is wrongly tagged.

Table 2.4: The rounds used before convergence and computing time for each model.

Model	Bookstore		User Tagging	
	Rounds	Time(s)	Rounds	Time (s)
Voting	1	0.2	1	0.5
2-Estimates	29	21.2	32	628.1
TruthFinder	8	11.6	11	435.0
Accu	22	185.8	23	3339.7
MSS	9	10.3	12	366.2

true values based on object clusters in the feature space. It has shown better accuracy compared with the original algorithms [18]. Table 2.1 shows the average precision and recall on the 12 tags by the compared algorithms. We can see that MSS still performs the best among these compared algorithms. The Figure 2.4(b) illustrates the scatter plot between the predicted reliability of each group and the average accuracy of its member sources on the test set. It is evident that the obtained group reliability is still a good predictor of the true accuracy with strong correlation coefficient 0.8676. This guarantees a competitive performance of the MSS model on this Flickr data set as on the book author data set.

We also compare the computational time used by different algorithms in Table 2.4. The experiments are conducted on a personal computer with Intel Core i7-2600 3.40 GHz CPU, 8 GB physical memory and Windows 7 operating system. We can see that compared with most of other algorithms,

MSS model can converge in fewer rounds with less computational cost.

2.9 Model Inference

In this section, we derive the variational inference for the proposed MSS model, and give the detail steps to update the variational parameters in each factor.

1: Update each factor $q(\boldsymbol{\pi}_{l,m}|\boldsymbol{\alpha}_{l,m})$ for the group observation parameter $\boldsymbol{\pi}_{l,m}$.

By variational approach, we can verify that the optimal $q(\boldsymbol{\pi}_{l,m}|\boldsymbol{\alpha}_{l,m})$ has the form

$$\begin{aligned} q(\boldsymbol{\pi}_{l,m}|\boldsymbol{\alpha}_{l,m}) &\propto \exp\{\mathop{\mathbb{E}}_{q(\boldsymbol{r}_{l,m}),q(t_m)} \ln p(\boldsymbol{\pi}_{l,m}|r_{l,m},t_m) \\ &\quad + \sum_{n\in I_{\cdot,m}} \mathop{\mathbb{E}}_{q(\boldsymbol{g}_n)} \ln p(y_{n,m}|\boldsymbol{\pi}_{l,m},g_n)\} \\ &\propto \prod_{k\in\mathcal{X}} {\pi_{l,m;k}}^{\alpha_{l,m;k}-1} \end{aligned}$$

It still has Dirichlet distribution with the parameters

$$\begin{aligned} \alpha_{l,m;k} = &\sum_{n\in I_{\cdot,m}} q(g_n = l)\delta\left[\!\left[y_{n,m} = k\right]\!\right] \\ &+ \sum_{r_{l,m}\in\{0,1\}} q(r_{l,m})[(\eta^{(r_{l,m})} - 1)q(t_m = k) \\ &+ (\theta^{(r_{l,m})} - 1)(1 - q(t_m = k))] + 1 \end{aligned}$$

for each $k \in \mathcal{X}_m$, where $\delta\left[\!\left[A\right]\!\right]$ is the indicator function which outputs 1 if A holds, and 0 otherwise. Here we index the element in $\boldsymbol{\alpha}_{l,m}$ and $\boldsymbol{\pi}_{l,m}$ by k after the colon.

2: Update each factor $q(u_l|\boldsymbol{\beta}_l)$ for general group reliability u_l.

We have

$$\begin{aligned} \ln q(u_l|\boldsymbol{\beta}_l) &\propto \sum_m \mathop{\mathbb{E}}_{q(r_{l,m})} \ln p(r_{l,m}|u_l) + \ln p(u_l|b_1,b_0) \\ &= (\sum_m q_1(r_{l,m}) + b_1 - 1)\ln u_l \\ &\quad + (\sum_m q_0(r_{l,m}) + b_0 - 1)\ln(1-u_l) \end{aligned}$$

where $q_i(r_{l,m})$ is short for $q(r_{l,m} = i)$ for $i = 0, 1$, respectively. It is evident the posterior of u_l still has beta distribution as Beta($\boldsymbol{\beta}_l$) with parameter

$$\boldsymbol{\beta}_l = [\sum_m q_1(r_{l,m}) + b_1, \sum_m q_0(r_{l,m}) + b_0]$$

It is evident that the above updated parameter sums up the posterior reliability $q_1(r_{l,m})$ and $q_0(r_{l,m})$ over all objects. This corresponds to the intuition that the general reliability is the sum of the reliability on individual objects.

3: Update each factor $q(r_{l,m}|\boldsymbol{\tau}_{l,m})$ for the object-specific reliability $r_{l,m}$ of group G_l on O_m:

$$\begin{aligned} \ln q(r_{l,m}|\boldsymbol{\tau}_{l,m}) \propto & \mathop{\mathbb{E}}_{q(t_m), q(\boldsymbol{\pi}_{l,m})} \ln p(\boldsymbol{\pi}_{l,m}|r_{l,m}, t_m) \\ & + \mathop{\mathbb{E}}_{q(u_l)} \ln p(r_{l,m}|u_l) \end{aligned} \tag{2.5}$$

Thus, we have

$$\begin{aligned} & \ln q(r_{l,m}|\boldsymbol{\tau}_{l,m}) \\ & \propto \sum_{k \in \mathcal{X}_m} q(t_m = k)[(\eta^{(r_{l,m})} - 1) \mathop{\mathbb{E}}_{q(\boldsymbol{\pi}_{l,m})} \ln \pi_{l,m;k} \\ & + (\theta^{(r_{l,m})} - 1) \sum_{j \neq k} \mathop{\mathbb{E}}_{q(\boldsymbol{\pi}_{l,m})} \ln \pi_{l,m;j}] \\ & + r_{l,m} \mathop{\mathbb{E}}_{q(u_l)} \ln u_l + (1 - r_{l,m}) \mathop{\mathbb{E}}_{q(u_l)} \ln(1 - u_l) \end{aligned} \tag{2.6}$$

for $r_{l,m} \in \{0, 1\}$, respectively. Here we compute the expectation of the logarithmic Dirichlet variable as

$$\mathop{\mathbb{E}}_{q(\boldsymbol{\pi}_{l,m})} \ln \pi_{l,m;k} = \psi(\alpha_{l,m;k}) - \psi(\sum_i \alpha_{l,m;i})$$

with the digamma function $\psi(\cdot)$; the expectation of the logarithmic Beta variables

$$\mathbb{E}_{q(u_l)} \ln u_l = \psi(\beta_{l;1}) - \psi(\beta_{l;1} + \beta_{l;2})$$

and

$$\mathbb{E}_{q(u_l)} \ln(1 - u_l) = \psi(\beta_{l;2}) - \psi(\beta_{l;1} + \beta_{l;2})$$

Finally, the updated values of $q(r_{l,m})$ are normalized to be valid probabilities.

The last line of Eq. (2.6) reflects how the general reliability u_l affects the estimation of the object-specific reliability. This embodies the idea that a generally reliable group is likely to be reliable on a particular object and vice versa. This can reduce the overfitting risk of estimating $r_{l,m}$ especially considering that $q(t_m)$ in the second line also needs to be estimated simultaneously in the *MSS* model as in the next step.

4: Update each factor $q(t_m|\boldsymbol{\nu_m})$ for the true value.

We have

$$\begin{aligned}&\ln q(t_m = k|\boldsymbol{\nu_m}) \propto \ln p(t_m = k)\\&+\sum_l \sum_{r_{l,m}\in\{0,1\}} q(r_{l,m}) \mathop{\mathbb{E}}_{q(\boldsymbol{\pi}_{l,m})} \ln p(\boldsymbol{\pi}_{l,m}|t_m = k, r_{l,m})\end{aligned}$$

This suggests that

$$\begin{aligned}&\ln q(t_m = k|\boldsymbol{\nu_m})\\&\propto \sum_l \sum_{r_{l,m}} q(r_{l,m})\{(\eta^{(r_{l,m})} - 1) \mathop{\mathbb{E}}_{q(\boldsymbol{\pi}_{l,m})} \ln \pi_{l,m;k}\\&+\sum_{k'\neq k} (\theta^{(r_{l,m})} - 1) \mathop{\mathbb{E}}_{q(\boldsymbol{\pi}_{l,m})} \ln \pi_{l,m;k'}\}\end{aligned}$$

All $q(t_m = k), k \in \mathcal{X}_m$ are normalized to ensure they are validate probabilities.

5: Update each factor $q(g_n|\boldsymbol{\varphi}_n)$ for the group assignment of each source.

We can derive

$$\begin{aligned}&\ln q(g_n = l|\boldsymbol{\varphi}_n)\\&\propto \mathop{\mathbb{E}}_{q(\boldsymbol{\rho})} \ln p(g_n = l|\boldsymbol{\rho}) + \sum_{m\in I_{n,\cdot}} \mathop{\mathbb{E}}_{q(\boldsymbol{\pi}_{l,m})} \ln p(y_{n,m}|\pi_{l,m}, g_n = l)\\&= \mathop{\mathbb{E}}_{q(\boldsymbol{\rho})} \ln p(g_n = l|\boldsymbol{\rho}) + \sum_{m\in I_{n,\cdot}} \mathop{\mathbb{E}}_{q(\boldsymbol{\pi}_{l,m})} \ln \pi_{l,m;y_{n,m}}\end{aligned}$$

This shows that $q(g_n = l|\boldsymbol{\varphi}_n)$ is a multinomial distribution with its parameter

as

$$\varphi_{n;l} = q(g_n = l|\boldsymbol{\varphi}_n) = \frac{\exp(U_{n,l})}{\sum_{l=1}^{\infty} \exp(U_{n,l})} \tag{2.7}$$

where

$$U_{n,l} = \mathop{\mathbb{E}}_{q(\boldsymbol{\rho})} \ln p(g_n = l|\boldsymbol{\rho}) + \sum_{m \in I_{n,\cdot}} \mathop{\mathbb{E}}_{q(\boldsymbol{\pi}_{l,m})} \ln \pi_{l,m;y_{n,m}}$$

As in [19], we truncate after L groups: the posterior distribution $q(\rho_i)$ after the level L is set to be its prior $p(\rho_i)$ from Beta$(1, \kappa)$; and all the expectations $\mathop{\mathbb{E}}_{q(\boldsymbol{\pi}_{l,m})} \ln \pi_{l,m;k}$ after L are set to:

$$\mathbb{E}_{q(\boldsymbol{\pi}_{l,m})} \ln \pi_{l,m;k} \quad = \mathop{\mathbb{E}}_{q(t_m), p(r_{l,m})} \{ \ \mathbb{E}[\ln \pi_{l,m;k} | r_{l,m}, t_m] \ \}$$

with the prior distribution $p(r_{l,m})$ defined as Section 2.4 for all $l > L$, respectively. The inner conditional expectation in the above is taken with respect to the probability of $\boldsymbol{\pi}_{l,m}$ conditional on $r_{l,m}$ and t_m. Similar to the family of nested Dirichlet process mixture in [19], this will form a family of nested priors indexed by L for the MSS model. Thus, we can compute the infinite sum in the denominator of Eq. (2.7) as:

$$\sum_{l=L+1}^{\infty} \exp(U_{n,l}) = \frac{\exp(U_{n,L+1})}{1 - \exp(\mathop{\mathbb{E}}_{\rho_i \sim \text{Beta}(1,\kappa)} \ln(1 - \rho_i))}$$

6: Update $q(\rho_i)$ in GEM construction.

Before the truncation level L, the posterior distribution $q(\rho_i) \sim \text{Beta}(\phi_{i,1}, \phi_{i,2})$ is updated as

$$\phi_{i,1} = 1 + \sum_{n=1}^{N} q(g_n = i), \quad \phi_{i,2} = \kappa + \sum_{n=1}^{N} \sum_{j=i+1}^{\infty} q(g_n = j)$$

2.10 Parameter Estimation

The model parameters $\Theta = \{\eta^{(0)}, \theta^{(0)}, \eta^{(1)}, \theta^{(1)}, b_1, b_0, \kappa\}$ can be estimated by maximizing the log-likelihood $\log \mathcal{L}(q)$ by the obtained factorization q with the constraints $\eta^{(1)} > \theta^{(1)}$ and $\eta^{(0)} \leq \theta^{(0)}$. Since we require $\eta^{(1)} > \theta^{(1)}$ *strictly*

holds, we usually impose $\eta^{(1)} \geq (1+\epsilon)\theta^{(1)}$ with a positive value of ϵ, i.e., $\eta^{(1)}$ is larger than $\theta^{(1)}$ with a margin ϵ. This ensures the strict inequality and improves numerical stability. In the algorithm, we set $\epsilon = 0.5$. Then, the parameter estimation problem becomes the following:

$$\begin{aligned}
&\Theta^{\star} = \arg\max_{\Theta} \mathcal{L}(q) \\
&s.t., 0 \leq \eta^{(0)} \leq \theta^{(0)}, \eta^{(1)} \geq (1+\varepsilon)\theta^{(1)} \geq 0, \\
&b_1, b_0, \kappa \geq 0
\end{aligned}$$

This constrained optimization problem can be solved by many off-the-shelf gradient-based constrained optimization solvers with the following gradients:

$$\begin{aligned}
\frac{\partial \mathcal{L}}{\partial \eta^{(r)}} &= \sum_{l,m,k \in \mathcal{X}_m} \{\psi(\eta^{(r)} + (K_m - 1)\theta^{(r)}) - \psi(\eta^{(r)}) \\
&+ \psi(\alpha_{l,m;k}) - \psi(\sum_i \alpha_{l,m;i})\}
\end{aligned}$$

$$\begin{aligned}
\frac{\partial \mathcal{L}}{\partial \theta^{(r)}} &= \sum_{k \in \mathcal{X}_m} \{\psi(\eta^{(r)} + (K_m - 1)\,\theta^{(r)}) - (K_m - 1)\psi(\theta^{(r)}) \\
&+ \sum_{k'} \psi(\alpha_{l,m;k'}) - (K_m - 1)\psi(\sum_i \alpha_{l,m;i})\}
\end{aligned}$$

for $r \in \{0, 1\}$.

$$\frac{\partial \mathcal{L}}{\partial b_1} = \sum_l \psi(b_1 + b_0) - \psi(b_1) + \psi(\beta_{l,1}) - \psi(\beta_{l,1} + \beta_{l,2})$$

$$\frac{\partial \mathcal{L}}{\partial b_0} = \sum_l \psi(b_1 + b_0) - \psi(b_0) + \psi(\beta_{l,2}) - \psi(\beta_{l,1} + \beta_{l,2})$$

$$\frac{\partial \mathcal{L}}{\partial \kappa} = \sum_i \psi(1 + \kappa) - \psi(\kappa) + \psi(\phi_{i,1} + \phi_{i,2}) - \psi(\phi_{i,2})$$

2.11 Conclusion

In this chapter, we propose an integrated true value inference and group reliability approach. Dependent sources which are grouped together, and their (general and specific) reliability is assessed at the group level. The true data values are extracted from the reliable groups so that the risk of overusing the observations from dependent sources can be minimized. The overall

approach is described by a probabilistic multi-source sensing model, based on which we jointly infer group reliability as well as the true values for objects a posterior given the observations from sources. The key to the success of this model is to capture the dependency between sources, and aggregate the collective knowledge at the group granularity. We present experimental results on two real data sets, which demonstrate the effectiveness of the proposed model over other existing algorithms.

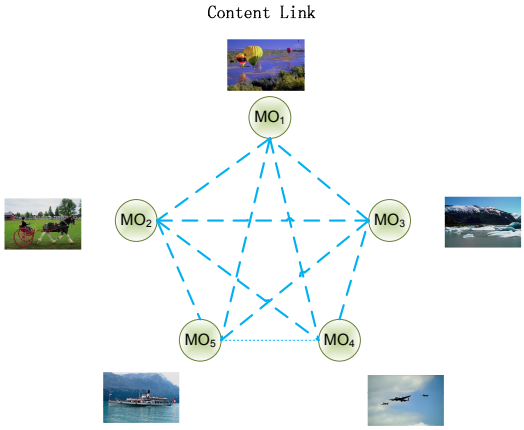
CHAPTER 3

INFORMATION INFERENCE

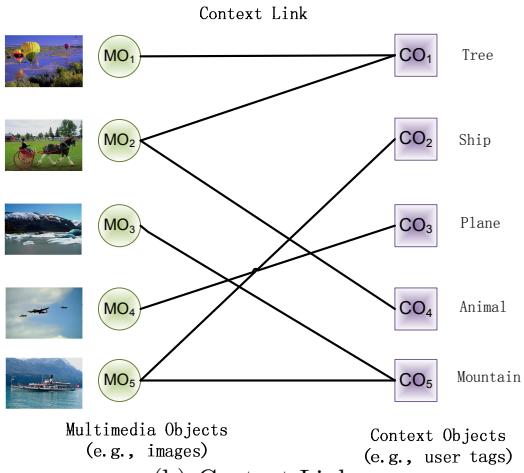
Social media networks contain both content and context-specific information. Most existing methods work with either of the two for the purpose of multimedia mining and retrieval. In reality, both content and context information are rich sources of information for mining, and the full power of mining and processing algorithms can be realized only with the use of a combination of the two. This chapter proposes a new algorithm, which mines both context and content links in social media networks to discover the underlying latent semantic space. This mapping of the multimedia objects into latent feature vectors enables the use of any off-the-shelf multimedia retrieval algorithms. Compared to the state-of-the-art latent methods in multimedia analysis, this algorithm effectively solves the problem of sparse context links by mining the geometric structure underlying the content links between multimedia objects. Specifically for multimedia annotation, we show that an effective algorithm can be developed to directly construct annotation models by simultaneously leveraging both context and content information based on latent structure between correlated semantic concepts. We conduct experiments on the Flickr data set which contains user tags linked with images. We illustrate the advantages of our approach over the state-of-the-art multimedia retrieval techniques.

3.1 Introduction

The development and popularity of Web 2.0 applications, has made it much easier for millions of users to create and share their personal multimedia objects (MOs) than ever before. Many image and video sharing websites have become extremely popular, as is evidenced by their burgeoning membership. Many such sites are built upon information and social network infrastructures



(a) Content Links



(b) Context Links

Figure 3.1: Context and content links in multimedia information networks.

such as Flickr, Youtube and Facebook that connect millions of users with one another. Users are able to share their multimedia objects with each other, and also provide the ability to tag each other's objects. Such sites represent a kind of rich multimedia information networks (MIN) [20] for social media [21],[22], in which the objects are linked to one another in the site with content links. By "content links," we refer to the visual and/or acoustic similarities between objects in a content feature space (see Figure 3.1(a)). At the same time, the sharing process of such sites naturally creates Context Objects (COs), because of the rich information provided by the different users directly or indirectly. Some examples of such context objects are tags (e.g., user tag and geo-tags), related attributes (colors, textures, and even categories from weakly labeled data) [23], and users who share MOs as well as their queries connected to multimedia objects by click-through records

(see Figure 3.1(b)). This helps create an even richer multimedia information network with context links, which connect the multimedia objects with their related context objects. For example, the multimedia objects clicked by users in the same query session probably contain the same semantic meaning. It is also the same for the multimedia objects which share the same user tags[1] in multimedia information networks. It is often very useful for multimedia retrieval by mining the semantics in these context links. In this chapter, we define a multimedia information network as an information network with two kinds of semantic objects – *multimedia objects* and *context objects*. See Figure 3.2 for an example. The multimedia objects are connected in a relational graph structure, with both content and context relationships. While content relationships are directly useful for retrieval, the context relationships also contain rich semantic information which should be leveraged for effective retrieval.

In this chapter, we show that a compact latent space can be discovered to summarize the semantic structure in multimedia information networks, which can be seamlessly applied in the state-of-the-art multimedia information retrieval systems (see Figure 3.2 for an example). Specifically, this algorithm maps each multimedia object into a latent feature vector that encodes the information in both context and content information. Based on these latent feature vectors, multimedia objects can be effectively classified, indexed and retrieved in a vector space by many mature off-the-shelf vector-based multimedia retrieval methods, like clustering, re-ranking [24] and Support Vector Machine (SVM) [25] for multimedia retrieval. Thus, our approach is a "general purpose technique," which can be leveraged to improve the effectiveness of a wide variety of techniques.

The general approach of learning latent semantic space has been extensively studied in the field of information retrieval. Popular techniques include Latent Semantic Indexing (LSI) [26], Probabilistic Latent Semantic Indexing (PLSI) [27] and Latent Dirichlet Allocation (LDA) [28]. These algorithms have also been applied to multimedia domain for problems such as indexing and retrieval [29], [30], [31] and [32]. For example, Bosch et al. [29] and

[1]In this chapter, we mainly concentrate on the context links associated with user tags. While the results in this chapter are general enough to be applied to any kind of context links, we mainly focus on tag links because of the richness of their semantic information as compared to other kinds of context links.

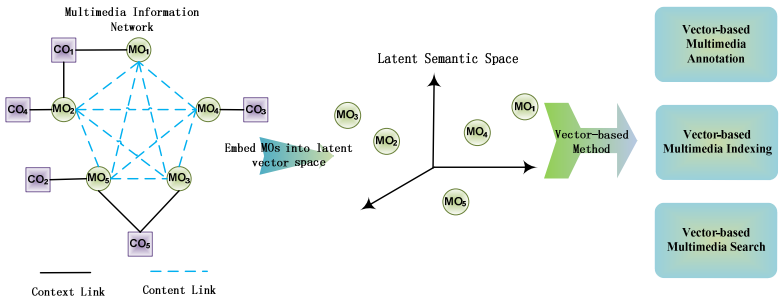


Figure 3.2: Learning latent semantic space from context as well as content links simultaneously

Monay et al. [30] learn latent feature vectors by LSI for natural scene images, and the learned features can be used effectively with general purpose SVM classifiers. Some preliminary results have shown the effectiveness of these algorithms, however, all these methods suffer from the problem with sparse context links, which we solve with the use of content links.

1. **Sparse Context Links.** These are the virtual links which are created as a result of user feedback (e.g., tags), and may be represented as the linkages between the multimedia objects and the contextual objects such as tags. In the real-world contextual links, the number of user tags attached to a multimedia object is usually quite small. In some extreme cases, only few or even no tag may be attached to an object, which often leads to sparse contextual links. In such cases, it is hard to derive meaningful latent features for multimedia objects, because the determination of the correlation structure in the latent space requires a sufficient number of such contextual objects to occur together.

   A reasonable solution to this problem is to exploit the content links between multimedia objects. In this chapter, we will show how the content links can effectively complement the sparse contextual links by incorporating acoustical and/or visual information to discover the underlying latent semantic space.

2. **Omitting Content Information in LSI Modeling of Context Links.** In this chapter, content links represent the content similarities between multimedia objects, i.e., those visually and/or acoustically similar objects are assumed to have strong content links between them.

Content links contain important knowledge complementary to that embedded in context links. However, to the best of our knowledge, the existing latent space methods, LSI, PLSI and LDA, cannot seamlessly incorporate the content and context links in a unified framework. Some attempts have been made to jointly model content and context information to learn the latent space [30], [32]. They quantify the multimedia objects into visual words, which are treated in the similar way as some context objects by linking them to multimedia objects. However, such approaches greatly increase the number of parameters in the latent space model, and make it more prone to quantization-induced noise and overfitting due to the sparse context links.

In contrast, we will show that content and context links can be seamlessly modeled to learn the underlying latent space. The content information does not have to be quantified into some discrete elements such as visual words described in [30]. Instead, the content link structure will be directly leveraged to discover latent features together with context links.

Therefore, we propose an elegant mapping of multimedia information networks to the latent space, which can support an emerging paradigm of multimedia retrieval which unifies the information in context and content links. In other words, the goal of this approach is to annotate the images with some manually defined concepts, using visual and contextual features for learning a latent space. Specifically, by feeding the latent vectors into existing classification models, it can be applied to multimedia annotation, which is one of the most important problems in multimedia retrieval. Furthermore, we show a more sophisticated algorithm, which can directly incorporate the discriminant information in training an example for multimedia annotation without using mapping as a pre-step. It jointly explores the context and content information based on a latent structure in the semantic concept space. Moreover, even given a new multimedia object with no context links, this extended algorithm can still annotate it. This solves the out-of-sample problem and greatly extends the applicability of the algorithm in multimedia retrieval applications.

3.1.1 Related Work

Analysis and inference with multi-modal data [33], [34], [35] have become one of most important research topics in computer vision and pattern recognition areas. Existing methods usually assume that in each data piece, there are a number of complementary cues associated with each other. For example, in a video clip, we observe a sequence of video frames as its visual cue, as well as the incident audio track. In the multi-modal problem, the data in different modalities is always associated with each other. In other words, one data modality is always associated with its counterparts in another modalities. Many representative works concentrate on this problem. SimpleMKL [35] addresses the multi-modal problem by learning a linear combination of multiple kernels with a weighted 2-norm formulation. Bekkerman and Jeon [33] explore the multi-modal nature of multimedia collections within the unsupervised learning framework. Guillaumin et al. [34] proposes to use semi-supervised learning to explore both labeled and unlabeled images in photo sharing websites while exploring the associated keywords in the text modality. Competitive results show these multi-modal algorithms can gain much better performance as compared with single modal algorithms.

However, in social media applications, content objects are not always associated with context objects. For example, the new images in a test set usually do not have any accompanying user tags. In this case, multi-modal methods cannot be applied due to the missing context objects. We will discover the missing links between context and content objects, which is one of main problems we will address in this chapter. In social media, structured multimedia information networks are the most natural data structure to represent the interaction between content and context objects. This chapter proposes a principled method to fuse the content and context objects in such a social media network structure. Specially, we attempt to capture the links in MIN by embedding the content objects into a latent space. Similar linear embedding techniques like metric learning [36] have been proposed to reveal the underlying space structure. However, it is nontrivial to extend these embedding techniques to MIN. Perhaps, the most relevant work is proposed by Blei et al. [28] who use a latent method for associating the annotated tags with the local regions in images. Its limitation is that this method can only assign existing user tags to images, but cannot handle the concepts beyond

these tags.

The remainder of this chapter is organized as follows. Section 3.2 reviews a set of state-of-the-art retrieval paradigms and unifies both context and content links in social media. In Section 3.3, we briefly review the basic ideas of latent methods which are closely related to the proposed method. The proposed latent method is then detailed in Section 3.4. In Section 3.5, we develop an advanced annotation model by exploring the context and content information with the latent structure between the correlated semantic concepts for annotation. Experimental results are presented in Section 3.6 on a real-world multimedia data set crawled from Flickr. Finally, conclusions are made in Section 3.7.

3.2 Exisiting Retrieval Paradigms

In the following, we briefly review some existing multimedia retrieval paradigms, and discuss the advantages of unifying analyses of both context and content links in social media. Based on whether context and/or content links are used, multimedia retrieval has evolved from the Content-based Multimedia Retrieval (CMR) [37] in the first paradigm, to the context-based multimedia retrieval (CxMR) in the second paradigm, and to the Context-and-Content-based Multimedia Retrieval (C2MR) as the most recent paradigm.

3.2.1 Content-Based Multimedia Retrieval

The CMR approach attempts to model the high-level concepts from the low-level concepts extracted from the multimedia objects. In a typical multimedia retrieval system like QBIC [38] and Virage [39], the query is formulated by some example multimedia objects and/or text-based keywords. Then, the relevant multimedia objects are retrieved based on their content features. The advantage of content-based multimedia retrieval (CMR) is that it is an automatic retrieval approach. Once the concepts are modeled, no human labels are required to maintain it. However, due to the technical limits of artificial intelligence and multimedia analysis, its accuracy is often too low to output satisfactory retrieval results due to the semantic gap between low-level content features and high-level semantics.

3.2.2 Context-Based Multimedia Retrieval

With the development of Web 2.0 infrastructures, rich context links are often connected to multimedia objects on the media-rich websites such as Flickr, Youtube and Facebook. In contrast to pure content information, these links provide extra semantic information to retrieve and index MOs in the web environment. For a simple example, the images of "sea" and "sky" have similar color features which are difficult to distinguish by similarity in content feature space. However, by leveraging the user tags in their context links and mapping them into a new latent space by LSI, PLSI and LDA, they can be distinguished with the semantics in their context objects. Context-based Multimedia Retrieval (CxMR) approaches have been widely used in many practical multimedia search engines such as Google Images, which utilize the context links such as surrounding text and user tags. Although the information in the context links is useful in many cases, they are often sparse and noisy. In some cases, it can lead to questionable performance, when the context contains much more irrelevant information to the mining process. This is often evident from the Google Image results when the images do not match the corresponding search at all.

3.2.3 Context-and-Content Multimedia Retrieval

Unifying the information in both context and content links is an appealing approach to solve the limits inherent in the two paradigms discussed above. Context links provide high-level semantic information which can be effective for resolving the ambiguity in the content feature space due to the semantic gap inherent in a pure content-based approach. Similarly, content links between multimedia objects can serve as regularization which can avoid the overfitting problem due to the sparse and noisy context links. The combination of two techniques provides the solution to effective multimedia retrieval in the rich Web 2.0 environment, which is so-called *Multimedia Retrieval 2.0*. This approach formulates multimedia retrieval by unifying the content and context-based approaches. As compared with the above existing multimedia retrieval systems, the advantages of our algorithm include:

1. We propose a general-purpose scheme which is broadly applicable. Many advanced vector-based retrieval systems can be seamlessly used

with the proposed approach.

2. Context and content links are explored in a unifying framework. Hence, the learned latent space ought to be more optimal than the other methods which separately mine these two kinds of links in multimedia information networks.

3. Specifically, for the multimedia annotation problem, a more sophisticated algorithm is developed by leveraging the assumption that the semantic concepts for annotation are correlated and thus a latent structure exists in such a semantic concept space. Also, the context-and-content links are simultaneously explored to optimize the annotation performance.

3.3 Latent Semantic Indexing

In this section, we briefly review latent semantic indexing, which is closely related to the algorithms proposed in this chapter. In conventional methods for LSI, we map MOs (multimedia objects) to latent feature vectors. Suppose we have n MOs $\{d_1, d_2, \cdots, d_n\}$ and m COs (context objects) $\{c_1, c_2, \cdots, c_m\}$ such as user tags. The context links between these n MOs and the m COs are denoted by an $n \times m$ matrix A. The elements $A_{i,j} \in \mathbb{R}^{n \times m}$ of this matrix represent the weights of context links, e.g., $A_{i,j} = 1$ if the jth CO is assigned to ith MO, or $A_{i,j} = 0$ otherwise. The goal of LSI is to construct a set of feature vectors $\{X_1, X_2, \cdots, X_n\}$ in a latent semantic space $\mathbb{R}^k$ to represent these multimedia objects. LSI performs a Singular Vector Decomposition (SVD) on the matrix A as follows:

$$A = U\Sigma V^T \tag{3.1}$$

Here, U and V are orthogonal matrices such that $U^TU = V^TV = I$, and the diagonal matrix Σ has the singular values as its diagonal elements. By retaining the largest k singular values in Σ and approximating others to be zero, LSI creates an approximated diagonal matrix $\widetilde{\Sigma}$ with fewer singular values. This diagonal matrix is used to approximate A as $\hat{A} = U\widetilde{\Sigma}V^T$. Then the matrix $X = U\widetilde{\Sigma} \in \mathbb{R}^{n \times k}$ yields a new feature representation, each row

of which is a k-dimensional feature vector of one multimedia object, i.e., $X = \left[\begin{array}{cccc} X_1 & X_2 & \cdots & X_n \end{array} \right]^T$. The computational complexity of SVD on the matrix A grows quadratically with the number of context objects. If the content features extracted from MOs are quantified into description words (e.g., visual words) as COs, the computational cost will increase rapidly. On the other hand, as stated in Section 3.1, the link matrix A is usually quite sparse with few context links. This may result in overfitting of the latent feature vectors, since the small number of context links may not reflect the underlying correlation structure in a robust way.

PLSI is another algorithm which models the latent space by context links. Each multimedia object is associated with a set of latent topic variables $\{h_1, h_2, \cdots, h_k\}$ with conditional probabilities $P(h_j|MO), 1 \leq j \leq k$. Similarly, for the latent topic h_l, the conditional probability of the context object CO_j is denoted by $P(CO_j|h_l)$. The conditional probability of CO_j given MO_i can be expressed as a product of these values:

$$P(CO_j|MO_i) = \sum_{l=1}^{k} P(CO_j|h_l) P(h_l|MO_i) \tag{3.2}$$

The probabilities $P(h_l|MO_i), P(CO_j|h_l), 1 \leq l \leq k$ can be estimated by using Maximum Likelihood (ML) and standard EM algorithms. We can use these to construct the latent feature vector $X(MO)$ of the multimedia object MO as follows:

$$X(MO) = [P(h_1|MO), P(h_2|MO), \cdots, P(h_k|MO)]^T \tag{3.3}$$

PLSI has similar drawbacks as LSI, because it does not consider the content links. Furthermore, the number of parameters in PLSI grows linearly with the number n of MOs. This suggests that the model is prone to overfitting [28] due to the sparse context links. Some alternative PLSI algorithms have been proposed for using context information during latent space discovery. They quantize the content features into COs (e.g., visual words) and use some extra conditional probabilities to model their relations with latent topics [32]. Although content information is used in such a model, it has many more parameters which need to be estimated. This results in overfitting.

LDA is another technique from this family of latent space methods. It

assumes that the probability distributions of multimedia objects over latent topics are generated from the same Dirichlet distribution [28]. This simplified assumption is key to avoiding the (large parameter) overfitting issue of PLSI. However, the simplifying assumption has the pitfall that the assumed Dirichlet distribution over MOs may not reflect their true distribution in the multimedia corpus.

While most of these algorithms focus on learning the latent space solely with context links, some efforts have been made to incorporate content information [40]. In order to incorporate content information into context analysis, it uses two separate matrices to factorize the content and context links (in addition to the latent matrix for multimedia objects). However, it does not consider the geometric structure of the distribution of multimedia objects in the corpus. From a practical perspective, the extra latent matrix for either content or context links is unnecessary in multimedia retrieval. Instead, in this chapter, we will learn a shared latent space from content and context links simultaneously, so that it can mine the link structure in an integrated manner without introducing any additional model parameters. Moreover, the proposed formulation has a better optimization topology, i.e., it is a global convex optimization problem so that better numerical stability can be achieved.

We propose to model the geometric structure of MOs by their content links to capture their distribution in the underlying latent space. In other words, our intuitive assumption is that *the MOs with stronger content links ought to be closer to each other in the latent space*. By this assumption, the content links can be encoded into latent space together with context links.

3.4 Latent Space Modeling in Social Media

In this section, we propose methods for combining the content links with context links in order to discover the latent semantic space for multimedia objects.

First, we show that the latent semantic indexing problem is closely related to low-rank matrix approximation [41], [42]. Due to the noises in the context

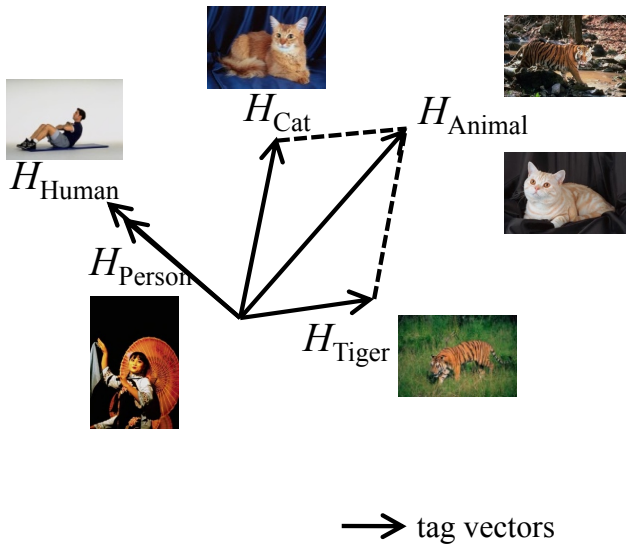


Figure 3.3: Illustration of latent low-rank structure among the tag vectors.

links, a noise term ε exist on the matrix A such that

$$A = H + \varepsilon \tag{3.4}$$

Here the matrix H denotes the noise-free context links, after the noise ε has been removed.

To derive H, some extra prior ought to be assumed on H. Inspired by LSI with a low-rank approximation of A, we impose a low-rank prior to recover H by minimizing the noisy term simultaneously as

$$\begin{aligned} &\min \|\varepsilon\|_F^2 + \gamma \text{rank}\,(H) \\ &s.t., A = H + \varepsilon \end{aligned} \tag{3.5}$$

where $\|\cdot\|_F$ is the Frobenius norm (i.e., the squared summation of all elements in a matrix), γ is the balancing parameter and rank$(\cdot)$ is the rank function.

There is an intuitive interpretation for the low-rank prior. Let $H_i, 1 \leq i \leq n$ denote the row vectors of H, which is the associated noise-free tag vector for the ith multimedia object. Each tag vector represents the occurrence of the corresponding tag in the multimedia corpus. As illustrated in Figure 3.3, the tag vectors of synonyms should be the same (or within a positive multiplier of one another), such as the tag vector H_{Person} and H_{Human} for the synonym terms "person" and "human." Moreover, many tags do not

independently occur in the corpus since they are semantically correlated. For example, the tag "animal" often correlates with its subclasses such as "cat" and "tiger." This indicates from the viewpoint of linear algebra, that the tag vector of "animal" could be located in a latent subspace spanned by those of its subclasses. Since the rank of matrix H is the maximum number of independent row vectors, it follows from the above dependency among tags, that H ought to have a low-rank structure. As revealed by the latent methods, user tags can be generated by mixing a few latent topics. The topic vectors that represent occurrences of the associated topics in the multimedia corpus span a latent semantic space, which contains most of tag vectors. Therefore, the rank of H should be no more than the maximum number of independent topic vectors in the latent space. Hence we can impose a low-rank prior to estimate the noise-free H from the observed noisy A.

It is NP-hard to directly solve the optimization problem of determining the lowest-rank approximation [41]. Recently, nuclear norm is proposed as a convex surrogate for matrix rank [43], [41]. Its convexity is an advantage in being able to perform an effective optimization process. The norm is computed as the sum of all the singular values of the matrix. Let $\|A\|_*$ denote the nuclear norm of A, then $\|A\|_* = \sum_i \sigma_i(A)$ where $\sigma_i(A)$ are singular values of A. Then Eq. (3.5) can be rewritten as

$$\min \|A - H\|_F^2 + \gamma \|H\|_* \tag{3.6}$$

The relationship between the above formulation and LSI can be presented more formally in the following result [41]:

Theorem 1. $\min_H \|A - H\|_F^2 + \gamma \|H\|_*$ *has a unique analytical solution as* $H_\gamma = U\mathrm{diag}\left(\left(\sigma - \frac{\gamma}{2}\right)_+\right)V^T$, *where* U, V *and* $\mathrm{diag}(\sigma)$ *form SVD for* A *as* $A = U\mathrm{diag}(\sigma)\mathrm{V}^{\mathrm{T}}$. *Here* $\mathrm{diag}(\sigma)$ *is a diagonal matrix with the singular values in vector* σ *such as its diagonal elements.* $(\sigma - \frac{\gamma}{2})_+$ *is a component-wise operation that* $(x)_+ = \max(0, x)$.

The difference is that LSI directly selects the largest k singular values of A but Eq. (3.6) subtracts $\frac{\gamma}{2}$ from each singular value and thresholds them by 0.

Suppose the resulting H is of rank k, then the SVD of H has the form of $H = U\Sigma_k V^T$ where Σ_k is a $k \times k$ diagonal matrix. Similar with LSI, the

row vectors of $X = U\Sigma_k$ can be used as the latent vector representations of multimedia objects in latent space. It is also worth noting that minimizing the rank of H gives a smaller k so that the obtained latent vector space can have lower dimensionality, and then the storage and computation in this space could be more efficient in practice.

However, Eq. (3.6) does not encode the content links, and the sparse context links may not result in a reliable latent space to represent multimedia objects. Suppose we are given a matrix Q of content links, where $Q_{i,j}$ can represent the similarity measurement between the ith MO and the jth MO. For example, we can extract some low-level feature vectors $\{\mathbf{f}_1, \mathbf{f}_2, \cdots, \mathbf{f}_n\}$ from the visual and/or acoustic content of MOs, then $Q_{i,j}$ could be represented as follows:

$$Q_{i,j} = \exp\left\{-\frac{\|\mathbf{f}_i - \mathbf{f}_j\|^2}{\sigma^2}\right\} \tag{3.7}$$

The relationship above uses a Gaussian kernel with radius σ.

By linking all the multimedia objects with Q, they can be embedded into a low-dimensional manifold structure [44], [45]. More specifically, we assume that *the multimedia objects with stronger links ought to be closer to each other in the latent semantic space*. This assumption is analogous to the Laplace-Beltrami operator on manifolds [44], and makes a smooth regularization on the underlying geometric structure between multimedia objects in the latent space. It can avoid the overfitting problem induced by sparse context links, and it can also incorporate the content links into modeling the latent space geometry. Based on this assumption, we introduce the quantity Ω to measure the smoothness of multimedia objects in the underlying latent space.

$$\begin{aligned}\Omega(X) &= \tfrac{1}{2}\sum_{i,j=1}^{n} Q_{i,j}||X_i - X_j||_2^2 \\ &= \tfrac{1}{2}\sum_{i,j=1}^{n} Q_{i,j}(X_i - X_j)(X_i - X_j)^T\end{aligned} \tag{3.8}$$

Here, $\|\cdot\|_2$ is l_2 norm, and X_i and X_j are the ith and jth row of X. It is easy to see that by minimizing the above regularization term, a pair of multimedia objects with larger $Q_{i,j}$ will have closer feature vectors X_i and X_j in the latent space. With some matrix operations, $\Omega(X)$ can be further

simplified as follows:

$$\begin{aligned}
&\Omega(X) = \tfrac{1}{2}\sum_{i,j=1}^{n} Q_{i,j}\left(X_i X_i^T - X_i X_j^T - X_j X_i^T + X_j X_j^T\right) \\
&= \sum_{i,j=1}^{n} Q_{i,j} X_i X_i^T - \sum_{i,j=1}^{n} Q_{i,j} X_i X_j^T \\
&= \text{trace}\left(XX^T D\right) - \text{trace}\left(XX^T Q\right) \\
&= \text{trace}\left(XX^T (D-Q)\right) \\
&= \text{trace}\left(X^T (D-Q) X\right) = \text{trace}\left(X^T L X\right)
\end{aligned} \tag{3.9}$$

Here, D is a diagonal matrix with its elements as the sum of each row of Q, and $L = D - Q$ is the positive semi-definite Laplacian matrix. By using the factorization $H = XV^T$ and $V^T V = I$, we can simplify as follows:

$$\begin{aligned}
&\text{trace}\left(H^T L H\right) = \text{trace}\left(V X^T L X V^T\right) \\
&= \text{trace}\left(X^T L X V^T V\right) = \text{trace}\left(X^T L X\right)
\end{aligned} \tag{3.10}$$

Now we can formulate the new model to discover the latent semantic space by plugging Eq. (3.10) into Eq. (3.6), which minimizes the following problem:

$$\min_H \mathcal{F}(H) = \|A - H\|_F^2 + \lambda \text{trace}\left(H^T L H\right) + \gamma \|H\|_* \tag{3.11}$$

Here λ is a balancing parameter. We note that the nuclear norm is convex, and L is a positive semi-definite matrix. Therefore, the above optimization problem has the desirable property that it is convex with a global optimum. Note that when there are images without any associated context objects (e.g., testing images with no user tags), the term of the least-square error in Eq. (3.11) is computed on the images with context objects. It is the matrix completion problem in [41]. In this case, the second term plays the role of sharing and connecting the context knowledge between tagged and untagged images by their visual similarities.

It is worth noting that no links are established between context objects in Eq. (3.11). The reason we do not consider these links is that in order to link the context objects (e.g., user tags), external knowledge is required to measure the similarity between them, such as WordNet and Google distance for linking textual user tags. Although these links can provide extra information, misleading knowledge may be introduced from the external resources, which do not comply with the visual evidence. For example, there is domain gap

between text and visual similarities, and two textual tags that are strongly correlated in text documents may not co-occur in images. Thus in the context of multimedia retrieval, we shall not incorporate context links in the formulation.

In contrast to Eq. (3.6), Eq. (3.11) does not have a closed-form solution. Fortunately, this problem can be solved by the proximal gradient method [46] which uses a sequence of quadratic approximations of the objective function in Eq. (3.11) in order to derive the optimal solution. We define $K(H) = \|A - H\|_F^2 + \lambda \text{trace}\left(H^T L H\right)$, and observe that $\mathcal{F}(H) = K(H) + \gamma \|H\|_*^2$ is a summation of the differentiable function K and the nuclear norm. This helps in defining the update step as well. Given $H_{\tau-1}$ in the last step $\tau-1$, it can be updated by solving the following optimization problem which quadratically approximates $\mathcal{F}(H)$ by Taylor expansion of $K(H)$ at $H_{\tau-1}$ [46]:

$$\begin{aligned} H_\tau &= \arg\min_H K(H_{\tau-1}) + \langle \nabla K(H_{\tau-1}), H - H_{\tau-1} \rangle \\ &+ \frac{\alpha}{2} \|H - H_{\tau-1}\|_F^2 + \gamma \|H\|_* \\ &= \arg\min_H \frac{\alpha}{2} \|H - G_\tau\|_F^2 + \gamma \|H\|_* \\ &+ K(H_{\tau-1}) - \frac{1}{2\alpha} \|\nabla K(H_{\tau-1})\|_F^2 \end{aligned} \tag{3.12}$$

Note that the last two terms in the rightmost side of Eq. (3.12) do not depend on H_τ so they can be ignored when minimizing w.r.t. H_τ. The values of G_τ and α in the above expression are defined as follows:

$$\begin{aligned} G_\tau &= H_{\tau-1} - \frac{1}{\alpha} \nabla K(H_{\tau-1}) \\ &= H_{\tau-1} - \frac{2}{\alpha} \left(H_{\tau-1} - A + \lambda L^T H_{\tau-1}\right) \end{aligned} \tag{3.13}$$

$$\alpha = 2\sigma_{\max}\left(I + \lambda L^T\right) \tag{3.14}$$

where the coefficient α satisfies the Lipschitz condition such that

$$\|\nabla_R K(R) - \nabla_T K(T)\|_F \leq \alpha \|R - T\|_F$$

for any R, T, and $\sigma_{\max}(\cdot)$ denotes the largest singular value.

In each step, (3.12) provides an analytical solution to H_τ, as illustrated in Theorem 1. Algorithm 1 summarizes the optimization procedure.

Algorithm 1 Proximal Gradient for Minimizing Eq. (3.11)

input A for the context links, Q for the content links, balance parameters λ and γ.

1 Initialize $H_0 \leftarrow 0$ and $\tau \leftarrow 1$.

2 Set $\alpha \leftarrow 2\sigma_{\max}\left(I + \lambda L^T\right)$.

repeat

2 Compute G_τ in Eq. (3.13).

3 Set $H_\tau \leftarrow U\mathrm{diag}\left(\sigma - \frac{\gamma}{\alpha}\right)_+ \mathrm{V}^{\mathrm{T}}$ which optimizes Eq. (3.12) by Theorem 1. Here $U\mathrm{diag}\left(\sigma\right)\mathrm{V}^{\mathrm{T}}$ gives the SVD of G_τ.

4 $\tau \leftarrow \tau + 1$.

until Convergence or maximum iteration number achieves.

3.5 Annotation Model with Context and Content Links

Multimedia annotation plays the critical role in multimedia retrieval, and it aims at annotating semantic concepts to multimedia objects. As already mentioned, once the latent feature vectors are learned, they can be fed into some existing vector-based classifiers to detect semantic concepts for annotation. Instead of learning a latent space for multimedia objects as a pre-step, we develop an alternative algorithm in this section that directly learns the annotation model from training examples. Our method explores both the context and content information based on the latent structure between the correlated semantic concepts for annotation. Since it is a supervised algorithm, we will refer to it as Supervised Context-and-Content Multimedia Retrieval (S-C2MR) in this chapter (in contrast to the U-C2MR algorithm). It is worth noting that even given a new multimedia object without any associated context links, S-C2MR can still annotate it. In other words, S-C2MR can readily handle the *out-of-sample problem* in the case of new multimedia objects. This greatly extends the applicability of content and context based multimedia annotation in many practical applications.

For a set of l semantic concepts, the goal of multimedia annotation is to predict the labels of these concepts on the multimedia objects. A set of n multimedia objects are used as the training data set to learn the annotation model, on which the labels of l concepts are given. Let $y_{i,u}$ denote the training label of the uth concept for the ith MO, where $y_{i,u} = +1$ denotes the positive label and $y_{i,u} = -1$ denotes the negative label. Meanwhile, a set of d-

dimensional raw feature vectors $\{\mathbf{f}_1, \mathbf{f}_2, \cdots, \mathbf{f}_n\}$ (e.g., the visual features for images and audio-visual features for videos) are extracted from the training set. To predict the labels, l linear classifiers are to be learned, where $W_u \in \mathbb{R}^d, u = 1, 2, \cdots, l$ are the coefficient vectors for these linear classifiers. Then, $\tilde{y}_{i,u} = W_u^T \mathbf{f}_i$ is the prediction score for the uth concept on the ith multimedia objects. Stacking W_u into a $d \times l$ matrix $W = [W_1, W_2, \cdots, W_l]$, $Y_i = W^T \mathbf{f}_i$ is the l-dimensional label vectors for all the l concepts on the ith multimedia object.

In the learning phase, we learn the model parameter W. The aim is to ensure that the prediction scores given by W should match with the ground truth labels on the training set as much as possible. Let $m_{i,u} = y_{i,u}\tilde{y}_{i,u} = y_{i,u}W_u^T \mathbf{f}_i$, then it should be as large as possible by the maximum margin principle. We use the logistic loss function $h_\theta(x) = \frac{1}{\theta}\log(1 + \exp(-\theta x))$ to measure the margin with θ controlling its shape, and the margin can be maximized by minimizing the total logistic loss over all the training examples:

$$\mathcal{L}(W) = \sum_{i=1}^{n}\sum_{u=1}^{l} h_\theta(m_{i,u}) = \sum_{i=1}^{n}\sum_{u=1}^{l} h_\theta\left(y_{i,u}W_u^T \mathbf{f}_i\right) \tag{3.15}$$

To incorporate the information from the context links, when learning W, we define an $n \times n$ symmetric matrix S, where each entry $S_{i,j}$ counts the number of context objects that the ith and the jth multimedia objects share. Actually, S can be computed as $S = AA^T$, and it summarizes the information in the context links. Similar to the smoothness assumption on the content links, it is also reasonable to assume that if two multimedia objects share more context objects, they ought to be semantically similar and the predicted label vectors on them should be as close as possible. Formally, this smoothness condition can be obtained by minimizing the following:

$$\begin{aligned}
&\Gamma(W) = \tfrac{1}{2}\sum_{i,j=1}^{n} S_{i,j}\left\|Y_i - Y_j\right\|_2^2 \\
&= \tfrac{1}{2}\sum_{i,j=1}^{n} S_{i,j}\left\|W^T\mathbf{f}_i - W^T\mathbf{f}_j\right\|_2^2 \\
&= W^T F(J - S)F^T W \\
&= W^T F K F^T W
\end{aligned} \tag{3.16}$$

Here, $F = [\mathbf{f}_1, \mathbf{f}_2, \cdots, \mathbf{f}_n]$ is the $d \times n$ data matrix with the raw feature

vectors as its columns, J is a diagonal matrix whose element is the sum of each corresponding row vector of S and $K = J - S$ is the Laplacian matrix for the context links in contrast to the Laplacian matrix L for the content links in Eq. (3.9). The third equality in the above equation can be derived in the similar manner to Eq. (3.9).

Similar to the tag vectors illustrated in Figure 3.3, the target semantic concepts for annotation will not appear independently. The correlation between these concepts implies that a linear dependency structure exists among the predictions of these concepts on the multimedia objects. In other words, these concepts form a low-dimensional latent space, in which these concepts are (linearly) dependent on each other. Since each column vector of W corresponds to the prediction coefficients for the associated concept, the linear dependent structure among concept predictions implies that W ought to be of low rank. Combining Eq. (3.15) and Eq. (3.16) together with the above latent assumption of concept space, we can solve W by minimizing

$$\sum_{i=1}^{n}\sum_{u=1}^{l} h_\theta\left(y_{i,u}W_u^T\mathbf{f}_i\right) + \eta\text{trace}\left(W^TFKF^TW\right) + \mu\left\|W\right\|_* \tag{3.17}$$

where η and μ are the balancing parameters. Again, this optimization problem can be solved by proximal gradient algorithm in the similar way as in Section 3.4. In detail, let us denote

$$B\left(W\right) = \sum_{i=1}^{n}\sum_{u=1}^{l} h_\theta\left(y_{i,u}W_u^T\mathbf{f}_i\right) + \eta\text{trace}\left(W^TFKF^TW\right) \tag{3.18}$$

then given the fixed $W^{(\tau-1)}$ at iteration $\tau-1$, Eq. (3.17) can be quadratically approximated by Taylor expanding $B(W)$ at $W^{(\tau-1)}$

$$\begin{aligned}
&P_\tau\left(W, W^{(\tau-1)}\right)\\
&= B\left(W^{(\tau-1)}\right) + \left\langle\nabla B\left(W^{(\tau-1)}\right), W - W^{(\tau-1)}\right\rangle\\
&+\tfrac{\alpha}{2}\left\|W - W^{(\tau-1)}\right\|_F^2 + \mu\left\|W\right\|_*\\
&= \tfrac{\alpha}{2}\left\|W - G^{(\tau)}\right\|_F^2 + \mu\left\|W\right\|_*\\
&+B\left(W^{(\tau-1)}\right) - \tfrac{1}{2\alpha}\left\|\nabla B\left(W^{(\tau-1)}\right)\right\|_F^2
\end{aligned} \tag{3.19}$$

where

$$G^{(\tau)} = W^{(\tau-1)} - \frac{1}{\alpha}\nabla B\left(W^{(\tau-1)}\right) \tag{3.20}$$

Here $\nabla B\left(W^{(\tau-1)}\right)$ is an $l \times n$ matrix which is the gradient of $B(W)$ at $W^{(\tau-1)}$.

$B(W)$ consists of two terms, and we compute their gradients respectively. Note that the first term of logistic loss is always differentiable, so we have

$$\begin{aligned} &\frac{\partial}{\partial W_u}\left(\sum_{i=1}^{n}\sum_{u=1}^{l} h_\theta\left(y_{i,u}W_u^T\mathbf{f}_i\right)\right) \\ &= \sum_{i=1}^{n} y_{i,u}h'_\theta\left(y_{i,u}W_u^T\mathbf{f}_i\right)\mathbf{f}_i \end{aligned} \tag{3.21}$$

where $h'_\theta(z) = \dfrac{-1}{1+e^{\theta z}}$ is the derivative of logistic loss function h at z. Denote M is an $n \times l$ matrix with each entry $M_{i,u} = y_{i,u}h'_\theta\left(y_{i,u}W_u^T f_i\right)$, we have the gradient w.r.t. W

$$\nabla\left(\sum_{i=1}^{n}\sum_{u=1}^{l} h_\theta\left(y_{i,u}W_u^T f_i\right)\right) = F \cdot M \tag{3.22}$$

Therefore, the gradient of $B(W)$ is

$$\nabla B(W) = F \cdot M + 2\eta FKF^T W \tag{3.23}$$

Then the new $W^{(\tau)}$ at iteration τ can be solved by

$$\begin{aligned} W^{(\tau)} &= \arg\min_W P_\tau\left(W, W^{(\tau-1)}\right) \\ &= \arg\min_W \tfrac{\alpha}{2}\left\|W - G^{(\tau)}\right\|_F^2 + \mu\|W\|_* \end{aligned} \tag{3.24}$$

which has analytical solution according to Theorem 1. Note that as pointed out in [46], the convergence of the proximal gradient algorithm can be accelerated by making an initial estimate of α (here, we initialize α by $\sigma_{\max}(\nabla B(W^{(\tau-1)}))$ in each iteration) and multiplying it by a constant factor ρ ($= 0.7$ in our case) until $B\left(W^{(\tau)}\right) + \mu\|W^{(\tau)}\|_* \leq P_\tau\left(W^{(\tau)}, W^{(\tau-1)}\right)$. Algorithm 2 summarizes the optimization process.

In the inference phase, given the raw feature vector $\mathbf{f}$ of a new multimedia object, its labels on l concepts can be predicted by $\tilde{y}(\mathbf{f}) = sign(W^T\mathbf{f})$.

Finally, we distinguish the proposed supervised content-and-context multimedia annotation algorithm from other latent models, including the one proposed in Section 3.4. Previous latent methods, such as latent semantic

Algorithm 2 Supervised Content-and-Context-Based Multimedia Annotation

input Matrix S, balance parameters η and μ.

1 Initialize $W^{(0)} \leftarrow 0$ and $\tau \leftarrow 1$.
repeat
2 Compute the gradient of $B(W)$ at $W^{(\tau-1)}$ as Eq. (3.23).
3 Set $G^{(\tau)} = W^{(\tau-1)} - \frac{1}{\alpha}\nabla B\left(W^{(\tau-1)}\right)$.
4 Set $W^{(\tau)} \leftarrow U\mathrm{diag}\left(\sigma - \frac{\mu}{\alpha}\right)_+ \mathrm{V}^{\mathrm{T}}$, where $U\mathrm{diag}(\sigma)\mathrm{V}^{\mathrm{T}}$ is the SVD of $G^{(\tau)}$.
8 $\tau \leftarrow \tau + 1$.
until Convergence or maximum iteration number achieves.

analysis [26], probabilistic latent semantic analysis [27] and latent Dirichlet allocation [28], are restricted to latent factor discovery. On the contrary, in this section, the goal of our approach is to directly model the semantic concepts from the content and context links while exploring their latent semantic correlations.

3.6 Experiments

To evaluate the proposed latent space method and its application in Context-and-Content-based Multimedia Retrieval (C2MR), we conduct experiments on a public multimedia data set with a large number of images as multimedia objects and noisy user tags as context objects. It is compared with the other paradigms of multimedia retrieval algorithms, such as Content-based Multimedia Retrieval (CMR) and Context-based Multimedia Retrieval (CxMR). We evaluate these algorithms in multimedia annotation problem, and their performances can be compared in quantity with the available labeling ground truth in the data set.

3.6.1 Data Set

Experiments are conducted on a publicly available Flickr data set[2]. It contains 55, 615 images which are crawled from the photo sharing website Flickr.com. The crawled images are linked to 1,000 user tags, which are

[2]http://lms.comp.nus.edu.sg/research/NUS-WIDE.htm

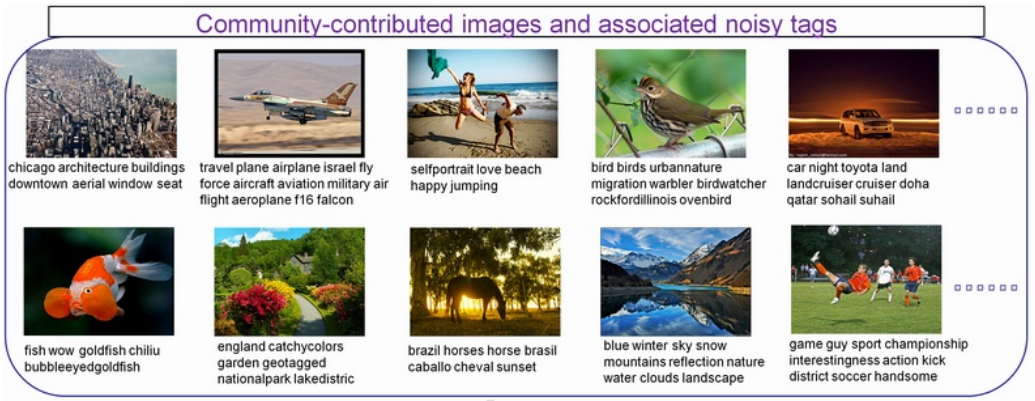


Figure 3.4: Examples of Flickr images and associated tags contributed by community.

annotated by users registered in Flickr. The context links between images and tags are quite sparse. In this data set, most of images only have fewer than 10 tags, and the average number of tags per image is 7.3. Figure 3.4 illustrates some example of images and their associated user tags.

Beyond these images and user tags, 81 concepts are defined in the data set for image annotation. Note that these 81 concepts are different from the user tags, and their ground truth labels are manually collected by the data set developer. In contrast, tags are annotated by amateur users in Flickr which contains many irrelevant noise information. The whole data set is partitioned into training set and test set for this annotation problem. The training set contains 27,807 images and the remaining 27, 808 images are in the test set. In the training set, the training labels are given for all 81 concepts to learn prediction model. The annotation performances are then evaluated on test set

Visual features extracted from the image corpus include the 64-D color histogram and 73-D edge direction histogram. These two kinds of features are concatenated together to form a 137-D vector feature [47]. Features are normalized by subtracting each dimension of feature by its mean, and then dividing the resulting feature by three times of the standard variation of this dimension. After that, the feature vectors of all samples are normalized so that the square sum of all the elements in each feature vector is one [47].

3.6.2 Performance Evaluation

The goal of multimedia retrieval is to retrieve a list which is relevant to the target concept. All the retrieved images are ranked according to their prediction scores in a descent order. The relevant images are expected to be ranked higher in the retrieved list. Therefore, to evaluate the ranking performance, we adopt Average Precision (AP) to measure the retrieval performance for each concept. Let R be the number of true positive images in the test set and R_j be the number of the relevant images in the top j images in the rank list. Let $I_j = 1$ if the jth image is relevant and 0 otherwise. Then AP is defined as

$$\frac{1}{R}\sum_{j}\frac{R_j}{j}I_j \tag{3.25}$$

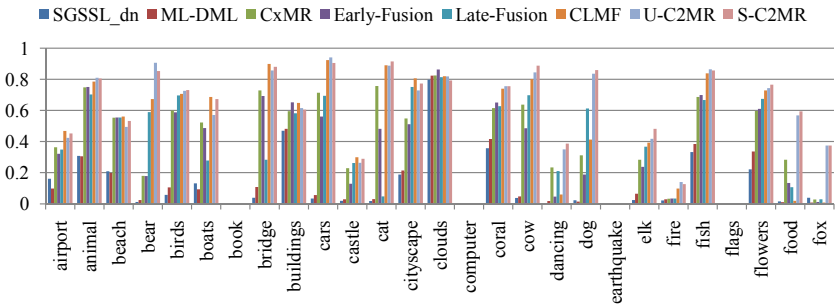
The AP corresponds to the area under a non-interpolated recall/precision curve and it favors highly ranked relevant images. In the experiments, AP is computed for each concept on the test set to measure the algorithm performance.

3.6.3 Comparison between Three Paradigms

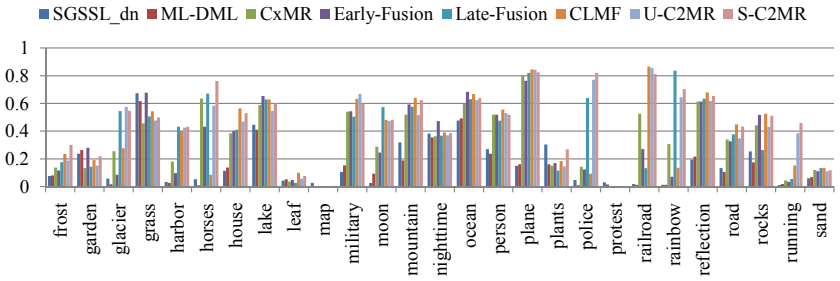
First, we compare the proposed algorithm with the other three paradigms of multimedia retrieval algorithms. For the sake of fair comparison, the SVM model is trained based on the learned latent space and/or visual features.

1. CMR - Content-based Multimedia Retrieval. Only visual features are used to model the 81 concepts. No user tags are used in this algorithm. In other words, we train SVM for each concept on visual features and the resulting SVM is used to predict the classification scores for retrieval. The Gaussian kernel is used in SVM for comparison.

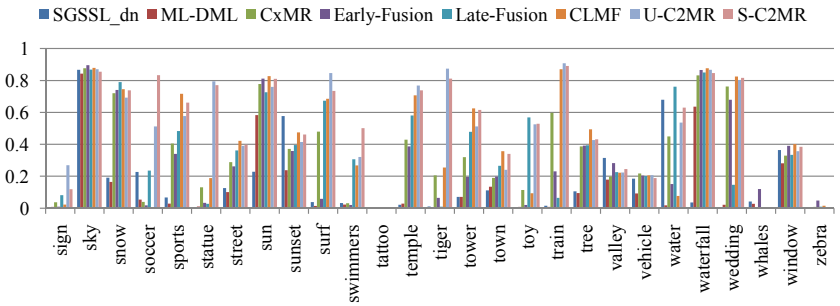
2. CxMR - Context-based Multimedia Retrieval. First, a latent space is learned solely from the context links between user tags and images based on PLSI. Then the SVM model is trained for each concept based on the obtained latent feature vectors to predict the scores. Next we will compare it with an advanced LSI variant - CLMF (i.e., combining Content and Link using Matrix Factorization [40]). We do not assume that user tags are available in the test set, thus in this paradigm of



(a) From "airport" to "fox"



(b) From "frost" to "sand"



(c) From "sign" to "zebra"

Figure 3.5: Comparison of different algorithms over 81 concepts on Flickr data set in terms of AP. The figure can be enlarged in the electric version.

latent methods, the user tags are predicated by their nearest neighbors in the training set.

3. C2MR - the proposed Context-and-Content-based Multimedia Retrieval. C2MR contains two different types - Unsupervised C2MR and Supervised C2MR.

    a. U-C2MR - Unsupervised C2MR. The algorithm in Section 3.4 is applied to model the latent space, which maps the multimedia objects into a latent space from both content and context links.

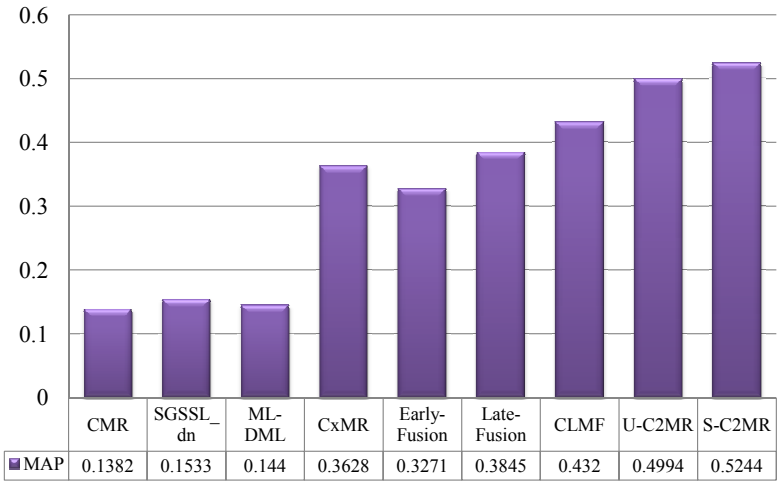


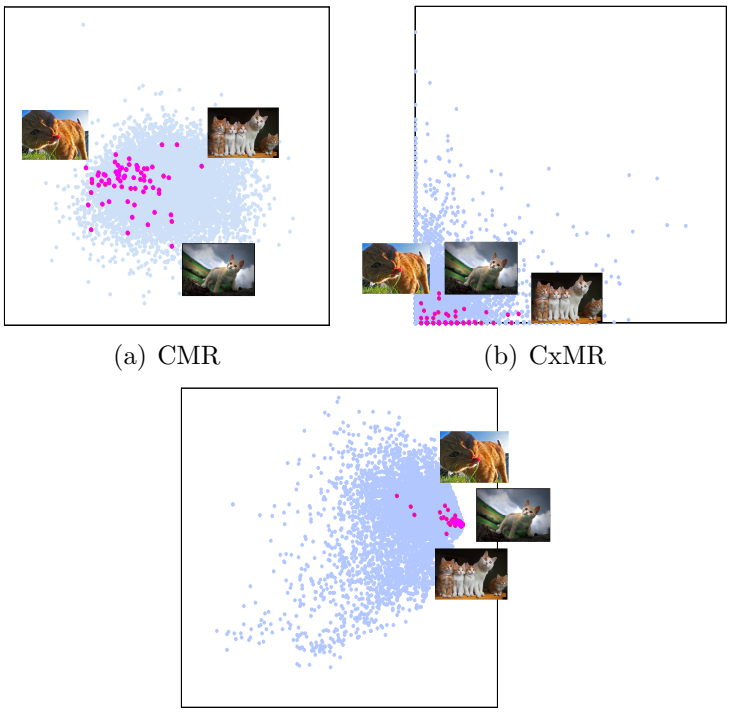
Figure 3.6: Comparison of different algorithms on 81 concepts on the Flickr data set in terms of MAP.

The parameters λ and γ in Eq. (3.10) are chosen from $\{0.2, 0.5, 1.0, 2.0\}$ via a 5-folder cross-validation on training set in terms of the resulting AP. Then, SVM is used to train classification models from the learned latent space.

b. S-C2MR - Supervised C2MR. The algorithm in Section 3.5 is developed for multimedia annotation. Different from U-C2MR, it directly learns classifier for the semantic concepts. The parameters η and μ in (3.17) are chosen from $\{0.2, 0.5, 1.0, 2.0\}$ via a 5-folder cross-validation on training set, and the shape parameter θ for the logistic loss is empirically set to be 1.0.

Figure 3.5 and Figure 3.6 illustrate the performances on all the compared algorithms. From the results, we have the following observations.

Among CMR, CxMR and C2MR, the proposed C2MR, both supervised and unsupervised versions, gain the best performances in terms of mean average precision (MAP) over all the 81 concepts. As for U-C2MR, it improves CMR by 246.8% and CxMR by 37.6%. Furthermore, S-C2MR improves CMR by 264.2% and CxMR by 44.5%. Meanwhile, of all 81 concepts, the proposed content and context multimedia retrieval methods (U-C2MR and S-C2MR) perform best on 58 concepts. On the remaining concepts, their performances only slightly deteriorate compared to the other algorithms.



(a) CMR (b) CxMR

(c) U-C2MR

Figure 3.7: Illustration of different algorithms of mapping of multimedia objects into a 2D latent space. The grey points correspond to the multimedia objects in the corpus, and the red ones correspond to those of "cat" images. (a) CMR: mapping multimedia objects into the 2D space by applying principal component analysis to visual features of images; (b) CxMR: mapping multimedia objects by PLSI into the 2D space; (c) U-C2MR: mapping multimedia objects into the 2D space by the proposed latent method in Section 3.4.

Comparing these three paradigms of multimedia retrieval methods, CMR performs worst since no semantic information in the user tag is used. CxMR performs much better than CMR, although the tag link is sparse and noisy. By regularizing the tag links by content links, C2MR significantly improves CxMR here. This is because by mining the similarity information in content links between MOs, visually similar Flickr images can implicitly "share" the tag links between each other, which relieves the problem with sparse tag links. On the other hand, the noise in tags can also be somewhat reduced in a latent semantic space by embedding context links and visual geometric structure in content links simultaneously.

Finally, we illustrate how different algorithms map multimedia objects into a 2D latent space in Figure 3.7. It shows that the proposed method

maps the multimedia objects with the same class (i.e., "cat" in this example) close to each other so that they have consistent feature representation in the underlying latent space. It gives an intuitive interpretation of better performance of the proposed algorithm, since it often becomes much easier to identify the region corresponding to a certain semantic class in the latent space, where the objects of this class are mapped together.

3.6.4 Comparison with Related Algorithms

We also compare the proposed algorithm with the other closely related algorithms.

1. Fusion – we combine the 137-D visual content features and the obtained context features in CxMR. The combined features are used to train the SVM model for each concept. There are the following two different fusion strategies - early-fusion and late-fusion [48].

    a. Early-Fusion: the two kinds of features are concatenated and directly fed into SVM to train the model for each concept.

    b. Late-Fusion: two SVM models are learned from visual and PLSI features respectively to predict scores for each concept, and the final prediction scores are given by linearly combining them in a late-fusion step.

2. SGSSL_dn – sparse graph-based semi-supervised learning approach together with handling tag noises [49]. In this algorithm, a concept space is explicitly constructed from the context links. Moreover, a sparse graph is constructed by datum-wise one-vs-kNN reconstructions of all samples, in which a training label refinement strategy is proposed to handle the noise in the user tags.

3. ML-DML – Multi-Label Distance Metric Learning [50]. This algorithm learns a semantic distance metric between visual features from user tags. Based on the learned distance, SVM is used to model each concept with a Gaussian kernel by exponentiating the obtained negative multi-label distance. Since it leverages user tags, it is compared with C2MR in the following.

4. CLMF – combining Content and Link using Matrix Factorization [40]. This algorithm combines the content and link analysis using matrix factorization. It attempts to symmetrically factorize the context matrix and asymmetrically factorize the content matrix. In this model, some extra latent variables are used to model the context topics.

By comparison in Figure 3.6, C2MR shows it can more effectively model the two links than the other fusion methods in terms of MAP. U-C2MR improves Early-Fusion by 52.7%, Late-Fusion by 35.3%, SGSSL_dn by 225.8%, and ML-DML by 247.0% and CLMF by 15.6%. S-C2MR improves Early-Fusion by 60.3%, Late-Fusion by 42.1%, SGSSL_dn by 242.1% and ML-DML by 264.2% and CLMF by 21.4%.

In fusion methods, late-fusion outperforms early-fusion. It indicates that simply concatenating context and content feature vectors together into a higher-dimensional vector cannot effectively utilize the context and content links. On the contrary, it is proven in the experiments that C2MR models a more informative latent space from the content and context links.

Finally, the comparison between ML-DML, SGSSL_dn and C2MR also shows C2MR can better utilize the information in the links of multimedia information networks. Although SGSSL_dn attempts to handle the noisy tags in context links, it does not solve the problem with sparse context links. Moreover, the concept space in this approach constructed from user tags is usually far from perfect due to the semantic gap. This makes it difficult to further improve the performance of multimedia retrieval built on this concept space. Although ML-DML also utilizes user tags to learn a discriminant metric structure in visual feature space, it does not explore the geometric structure in either content links as U-C2MR or the context links as S-C2MR. Moreover, it does not look into the intrinsic latent space of either the tag vectors as U-C2MR or the label vectors of semantic concepts as S-C2MR.

Although CLMF attempts to incorporate content information into context analysis, it uses two matrices to separately factorize the context and content links. On the contrary, the proposed model learns a shared latent matrix H from content and context links simultaneously. Indeed, from the practical perspective, one extra matrix for either content or context links is unnecessary in multimedia retrieval, and it needs extra training samples to learn a satisfactory model. With more compact latent structure, the proposed

Table 3.1: Comparison of computing time (in seconds) by latent methods and the other related methods.

	Algorithms	Computing Time
Latent Methods	CMR	N/A
	CxMR	8152.50 secs
	CLMF	3045.31 secs
	U-C2MR	2347.78 secs
	S-C2MR	3749.48 secs
Other Methods	SGSSL_dn	22680.0 secs
	ML-DML	349.57 secs

algorithm is more compact than CLMF with the shared latent matrix and thus has better performance as shown in the experiment. Moreover, the proposed model can reduce the noise-induced uncertainty by low-rank prior, and the sparse context links are complemented by embedding multimedia objects into their content linkage structure.

3.6.5 Comparison between U-C2MR and S-C2MR

Finally, we compare U-C2MR and S-C2MR. As shown in Figure 3.6, S-C2MR performs slightly better than U-C2MR by 5% improvement. The reason is that S-C2MR aims at directly learning the semantic concepts for annotation in a unified framework and it utilizes extra discriminant information to learn the corresponding model for the target concepts.

3.6.6 Computing Time

Experiments are conducted on a platform with Intel Xeon CPU 2.80GHz and 8G physical memory. Table 3.1 illustrates the computing time of d-ifferent algorithms compared above. Since CMR is conducted directly on low-level feature space without modeling the latent space, its computing time is not listed. By comparison, both U-C2MR and S-C2MR are more computationally efficient than CxMR and SGSSL_dn, and have the similar computation load with CLMF. On the other hand, although U-C2MR and S-C2MR perform more slowly than ML-DML, they improve the performance of ML-DML significantly in terms of MAP.

3.7 Conclusion

In this chapter, we propose an algorithm which discovers the latent semantic space from both context and content links in multimedia information networks. The algorithms solve the problem with sparse context links by enriching the multimedia information networks with content links, and multimedia objects are embedded into a geometric structure underlying their content information. We extend the traditional latent semantic indexing algorithm by low-rank approximation, in which the information from the content links is seamlessly incorporated. The learned latent semantic space can be applied for many applications, such as multimedia annotation and retrieval. Specifically, we develop a context-and-content-based multimedia annotation algorithm which can learn the concept models from the context links and content links simultaneously based on the intrinsic low-rank structure in the latent concept space. For evaluation, we compare the proposed algorithm with other multimedia retrieval paradigms with either content or context links on a real-world Flickr data set. Other related algorithms in multimedia information networks are compared as well. The results show that the proposed algorithm is quite effective to integrate the content and context links for semantic retrieval over all 81 concepts from Flickr data set.

CHAPTER 4

INFORMATION TRANSFER

The problem of transfer learning has recently been of great interest in a variety of machine learning applications. In this chapter, we examine a new angle to the transfer learning problem, where we examine the problem of distance function learning. Specifically, we focus on the problem of how our knowledge of distance functions in one domain can be transferred to a new domain. A good semantic understanding of the feature space is critical in providing the domain-specific understanding for setting up good distance functions. Unfortunately, not all domains have feature representations which are equally interpretable. For example, in some domains such as text, the semantics of the feature representation are clear, as a result of which it is easy for a domain expert to set up distance functions for specific kinds of semantics. In the case of image data, the features are semantically harder to interpret, and it is harder to set up distance functions, especially for particular semantic criteria. In this chapter, we focus on the problem of transfer learning as a way to close the semantic gap between different domains, and show how to use correspondence information between two domains in order to set up distance functions for the semantically more challenging domain.

4.1 Introduction

The problem of transfer learning [51], [52], [53], [32], [54] has seen a revival in recent years because of the tremendous amount of heterogeneous data which is available in a wide variety of networks and content-based applications. Different domains provide a different level of ease in data collection and processing. Therefore, it is useful to somehow transfer the knowledge from one domain to the other. For example, in cross-lingual learning, labeled English text is widely available, whereas it is much harder to obtain labeled

Chinese documents. Therefore, the focus of transfer learning in this example is to use the natural correspondence between the feature spaces of the two domains in order to create an automated learner for Chinese documents. The focus of most transfer learning problems is on aspects which involve the *unavailability of sufficient data* for learning purposes. The transfer learning model is used as a way to learn cases in which sufficient data is not available to create the classification model.

In this chapter, we examine a different angle to the transfer learning problem, by exploring the varying semantic gap [55] in different feature spaces. An understanding of the semantics of a feature space is critical in setting up key operations in that space. One such example is the problem of distance function design. Distance function design is a key problem for many fundamental applications such as similarity search [56], [36], [57], [58], [59], [60] and retrieval [50].

Distance functions can be set up much more easily in a feature space, when the semantics of that space are easy to interpret. This is especially true for applications in which the distance function needs to be designed with specific criteria in mind. For example, in the text domain, a distance function which is discriminatory between certain kinds of topics can be easily set up by restricting the feature space to words which belong to the set of topics at hand. On the other hand, this is much harder to achieve in a domain such as image data in which the features cannot be naturally interpreted in terms of the different criteria, and the distance function design is far more challenging.

In this chapter, we focus on the problem of transfer learning as a way to link the different domains. As an example, we assume that the only input to the process is a set of images with corresponding text in the learning phase. We would like to explore this correspondence between the two domains in order to set up a distance function which uses only the image features, even in a different collection of images which do not have corresponding text. We also note that in some cases, the metric information in original target domain may be available in order to further improve the accuracy of the transfer learning process.

The remainder of this chapter is organized as follows. In Section 4.2, we formally define the problem of transfer learning of distance functions across heterogeneous domains. Section 4.3 formulates and solves the optimization problem of learning the distance function through a transfer learning process.

We relate the proposed method with existing work in the literature in Section 4.4. In Section 4.5, we present experiments on real-world data sets and show the advantages of the proposed algorithm. Section 4.6 proves Theorem 2, and the conclusions are presented in Section 4.7.

4.2 Problem Definition and Target Metric

Let $\mathbb{R}^s$ and $\mathbb{R}^t$ be the source and target feature spaces with dimensionalities of s and t respectively. Each instance in the source space is represented by a feature vector $\mathbf{y} \in \mathbb{R}^s$, and the target instances are represented by feature vectors $\mathbf{x}$ in the target space $\mathbb{R}^t$. In order to transfer the metric structure from source domain to target domain, we define a random variable $\mathbb{1}_{\text{Rel}}(\mathbf{x}, \mathbf{y})$ to indicate the cross-domain relevance between a target instance $\mathbf{x}$ and a source instance $\mathbf{y}$. We define a transfer function $T(\mathbf{x}, \mathbf{y})$ to measure the probability of $\mathbf{x}$ and $\mathbf{y}$ being relevant to each other, over $\mathbb{R}^s \times \mathbb{R}^t$ as

$$T : \mathbb{R}^s \times \mathbb{R}^t \to [0, 1], (\mathbf{x}, \mathbf{y}) \mapsto T(\mathbf{x}, \mathbf{y}) \tag{4.1}$$

Then the cross-domain relevance variable $\mathbb{1}_{\text{Rel}}(\mathbf{x}, \mathbf{y})$ follows the Bernoulli distribution $\mathbb{B}(T(\mathbf{x}, \mathbf{y}))$ parameterized by the transfer function, i.e.,

$$p(\mathbb{1}_{\text{Rel}}(\mathbf{x}, \mathbf{y}) = 1) = T(\mathbf{x}, \mathbf{y})$$

and

$$p(\mathbb{1}_{\text{Rel}}(\mathbf{x}, \mathbf{y}) = 0) = 1 - T(\mathbf{x}, \mathbf{y})$$

Additionally, to capture the metric structure in source domain, the source space may use a particular kind of similarity function, which is the most effective for processing in that domain. For example, the cosine similarity function is likely to be quite effective in the text domain. We use a kernel function $k(\mathbf{y}, \tilde{\mathbf{y}})$ in order to encode this metric structure in the source space, which measures the similarity of $\mathbf{y}$ and $\tilde{\mathbf{y}}$ in the source space. Any Mercer kernel which satisfies the positive semi-definite property [25] in source space can be used here. In the meantime, we assume all the source instances are sampled from a true distribution $p(\mathbf{y})$. Then the kernel similarity together with $p(\mathbf{y})$ completely describes the metric structure between source instances.

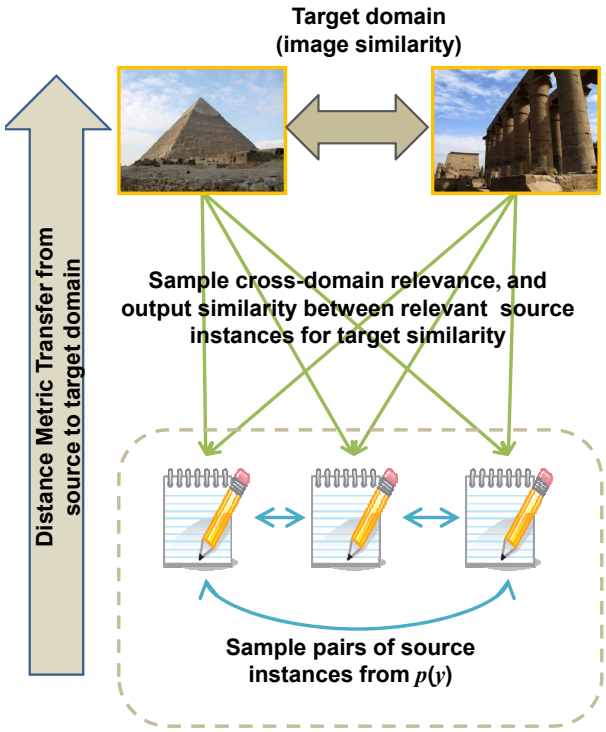


Figure 4.1: Illustration of computing the target image similarity from the relevant text documents sampled from the cross-domain metric sampling process. Although the pyramid (the left) and Luxor Temple (the right) images look visually different, both of them are semantically related in the context of text documents introducing Egyptian architecture.

Now given the kernel structure in *source* space, with the help of transfer function T we can define the metric structure in target space by exploring the metric structure in source space. Specifically, we depict the following cross-domain metric sampling process to compute the similarity between the target instances $\mathbf{x}$ and $\tilde{\mathbf{x}}$:

1. Sampling a pair of source instances $\mathbf{y}$ and $\tilde{\mathbf{y}}$ from $p(\mathbf{y})$.

2. Sampling $\mathbb{1}_{\text{Rel}}(\mathbf{x}, \mathbf{y}) \sim \mathbb{B}(T(\mathbf{x}, \mathbf{y}))$ and $\mathbb{1}_{\text{Rel}}(\tilde{\mathbf{x}}, \tilde{\mathbf{y}}) \sim \mathbb{B}(T(\tilde{\mathbf{x}}, \tilde{\mathbf{y}}))$ to decide whether $\mathbf{y}$ and $\tilde{\mathbf{y}}$ are relevant to $\mathbf{x}$ and $\tilde{\mathbf{x}}$, respectively.

3. If both are relevant, i.e., $\mathbb{1}_{\text{Rel}}(\mathbf{x}, \mathbf{y}) \cdot \mathbb{1}_{\text{Rel}}(\tilde{\mathbf{x}}, \tilde{\mathbf{y}}) = 1$, outputting $k(\mathbf{y}, \tilde{\mathbf{y}})$ as the target similarity between $\mathbf{x}$ and $\tilde{\mathbf{x}}$; otherwise, outputting 0 which means that in terms of the sampled source instances $\mathbf{y}$ and $\tilde{\mathbf{y}}$ no evidence shows the target instances $\mathbf{x}$ and $\tilde{\mathbf{x}}$ are similar.

Based on the above sampling process, we define the target similarity as

the *expected* output of the target similarity over $p(\mathbf{y})$:

$$\begin{aligned} & s\left(\mathbf{x}, \tilde{\mathbf{x}}\right) \\ & \triangleq \mathbb{E}_{\mathbf{y}, \tilde{\mathbf{y}} \sim p(\mathbf{y})}\left[\mathbb{E}\left[\mathbb{1}_{\mathrm{Rel}}(\mathbf{x}, \mathbf{y}) \cdot \mathbb{1}_{\mathrm{Rel}}(\tilde{\mathbf{x}}, \tilde{\mathbf{y}}) k(\mathbf{y}, \tilde{\mathbf{y}}) | \mathbf{y}, \tilde{\mathbf{y}}\right]\right] \\ & = \mathbb{E}_{\mathbf{y}, \tilde{\mathbf{y}} \sim p(\mathbf{y})}\left[T\left(\mathbf{x}, \mathbf{y}\right) T\left(\tilde{\mathbf{x}}, \tilde{\mathbf{y}}\right) k\left(\mathbf{y}, \tilde{\mathbf{y}}\right)\right] \\ & = \int_{\Delta \times \Delta} T\left(\mathbf{x}, \mathbf{y}\right) T\left(\tilde{\mathbf{x}}, \tilde{\mathbf{y}}\right) k\left(\mathbf{y}, \tilde{\mathbf{y}}\right) p(\mathbf{y}) p(\tilde{\mathbf{y}}) d\mathbf{y} d\tilde{\mathbf{y}} \end{aligned} \tag{4.2}$$

where Δ is the support of the distribution $p(\mathbf{y})$. It computes the target similarity metric by taking expectation of the source similarity $k\left(\mathbf{y}, \tilde{\mathbf{y}}\right)$ transferred by T with respect to $p(\mathbf{y})$.

Figure 4.1 illustrates this idea by demonstrating how (target) image similarity is computed from the relevant (source) text documents. The images are linked to the relevant text documents by sampling the cross-domain relevance variables. The transfer function is used to link the images to the relevant text documents. Then the target similarity between images is obtained by accumulating the similarities of the relevant text documents weighted by the transfer function. If the two text documents are relevant to the target images based on sampled relevance indicator variables, their similarity will be accumulated for computing the image similarity; otherwise the text similarity will be neglected since they describe irrelevant content to the images.

It can be proved that the above target similarity is a valid Mercer kernel function, which is the positive semi-definite by the Mercer theorem:

Theorem 2. *Given a positive semi-definite source kernel k, $s(\mathbf{x}, \tilde{\mathbf{x}})$ in Eq. (4.2) is a valid Mercer kernel.*

Proof. We show that s is a positive semi-definite kernel. For a set of finite target instances $\{\mathbf{x}_i, 1 \leq i \leq l\}$ and corresponding coefficients $\{\alpha_i, 1 \leq i \leq l\}$, we have

$$\begin{aligned} & \sum_{i,j=1}^{l} \alpha_i \alpha_j s\left(\mathbf{x}_i, \mathbf{x}_j\right) = \int_{\Delta \times \Delta} \sum_{i,j=1}^{l} \alpha_i \alpha_j T\left(\mathbf{x}_i, \mathbf{y}\right) T\left(\mathbf{x}_j, \tilde{\mathbf{y}}\right) \\ & \cdot k\left(\mathbf{y}, \tilde{\mathbf{y}}\right) p(\mathbf{y}) p(\tilde{\mathbf{y}}) d\mathbf{y} d\tilde{\mathbf{y}} = \int_{\Delta \times \Delta} \left(\sum_{i=1}^{l} \alpha_i T\left(\mathbf{x}_i, \mathbf{y}\right) p(\mathbf{y})\right) \\ & \cdot \left(\sum_{i=1}^{l} \alpha_i T\left(\mathbf{x}_i, \tilde{\mathbf{y}}\right) p(\tilde{\mathbf{y}})\right) k\left(\mathbf{y}, \tilde{\mathbf{y}}\right) d\mathbf{y} d\tilde{\mathbf{y}} \\ & = \int_{\Delta \times \Delta} \beta(\mathbf{y}) \beta(\tilde{\mathbf{y}}) k\left(\mathbf{y}, \tilde{\mathbf{y}}\right) d\mathbf{y} d\tilde{\mathbf{y}} \geq 0 \end{aligned} \tag{4.3}$$

where $\beta(\mathbf{y}) = \sum_{i=1}^{m} \alpha_i T(\mathbf{x}_i, \mathbf{y})\, p(\mathbf{y})$ and the last inequality follows from the semi-definite positivity of the kernel k. Thus $s(\mathbf{x}, \tilde{\mathbf{x}})$ is a valid Mercer kernel. □

According to the definition of the Mercer kernel, there exists a function $\phi(\mathbf{x})$ that maps each target instance $\mathbf{x}$ to $\phi(\mathbf{x})$ in an output feature space, in which the inner product is implicitly given by $s(\mathbf{x}, \tilde{\mathbf{x}}) = \langle \phi(\mathbf{x}), \phi(\tilde{\mathbf{x}}), \rangle$. Hence, the (squared) distance between two target instances can be computed as

$$\begin{aligned} & d_{\text{tgt}}(\mathbf{x}, \tilde{\mathbf{x}}) = \langle \phi(\mathbf{x}) - \phi(\tilde{\mathbf{x}}), \phi(\mathbf{x}) - \phi(\tilde{\mathbf{x}}) \rangle \\ & = s(\mathbf{x}, \mathbf{x}) + s(\tilde{\mathbf{x}}, \tilde{\mathbf{x}}) - 2s(\mathbf{x}, \tilde{\mathbf{x}}) \end{aligned} \tag{4.4}$$

This distance function formally satisfies the mathematical properties of a *metric*, i.e., this distance metric in the target space is symmetric, non-negative and satisfying the triangle inequality.

We define the target similarity in terms of a population expectation w.r.t. the true distribution $p(\mathbf{y})$ in Eq. (4.2). However, in reality the underlying $p(\mathbf{y})$ is unknown beforehand. Alternatively, we can consider the empirical version of the *true* target similarity. Given a set of source instances $\mathbf{y}_i, 1 \leq i \leq n$ i.i.d. sampled from $p(\mathbf{y})$, the empirical distribution is

$$p_n(\mathbf{y}) = \frac{1}{n} \sum_{i=1}^{n} \delta[\mathbf{y} - \mathbf{y}_i]$$

with the Dirac's delta function $\delta[\cdot]$. Substituting $p(\mathbf{y})$ with $p_n(\mathbf{y})$, we obtain the following *empirical* target similarity

$$\begin{aligned} & s_n(\mathbf{x}, \tilde{\mathbf{x}}) \\ & = \int_{\Delta \times \Delta} T(\mathbf{x}, \mathbf{y})\, T(\tilde{\mathbf{x}}, \tilde{\mathbf{y}})\, k(\mathbf{y}, \tilde{\mathbf{y}})\, p_n(\mathbf{y}) p_n(\tilde{\mathbf{y}}) d\mathbf{y} d\tilde{\mathbf{y}} \\ & = \frac{1}{n^2} \sum_{i,j=1}^{n} \{ T(\mathbf{x}, \mathbf{y}_i)\, T(\tilde{\mathbf{x}}, \mathbf{y}_j)\, k(\mathbf{y}_i, \mathbf{y}_j) \} \end{aligned} \tag{4.5}$$

Note that in the cross-domain metric sampling process the *pairs* of source instances are sampled independently. However, in $s_n(\mathbf{x}, \tilde{\mathbf{x}})$ the pairs of $(\mathbf{y}_i, \mathbf{y}_j)$ are not statistically independent although the $\mathbf{y}_i$'s are independently sampled from $p(\mathbf{y})$. The conventional analysis tools for i.i.d. samples do not apply in this case, and instead we apply the McDiarmid inequality [61], [62] to bound the difference between the true and empirical target similarity. We show that

$s_n(\mathbf{x}, \tilde{\mathbf{x}})$ asymptotically converges to $s(\mathbf{x}, \tilde{\mathbf{x}})$ at rate $O(\frac{1}{\sqrt{n}})$:

Theorem 3. *Given any two target instances* $\mathbf{x}$ *and* $\tilde{\mathbf{x}}$*, with probability at least* $1 - \mu$*, we have*

$$|s_n(\mathbf{x}, \tilde{\mathbf{x}}) - s(\mathbf{x}, \tilde{\mathbf{x}})| \leq \frac{1}{n} |\varrho(\mathbf{x}, \tilde{\mathbf{x}})| + B\sqrt{\frac{2}{n} \ln \frac{2}{\mu}} \tag{4.6}$$

where B *is the upper bound of the kernel function, i.e.,* $|k(\mathbf{y}, \mathbf{z})| < B$ *for any* $\mathbf{y}$ *and* $\mathbf{z}$*; and*

$$\varrho(\mathbf{x}, \tilde{\mathbf{x}}) = \mathbb{E}_{\mathbf{y} \sim p(\mathbf{y})} [T(\mathbf{x}, \mathbf{y}) T(\tilde{\mathbf{x}}, \mathbf{y}) k(\mathbf{y}, \mathbf{y})] - s(\mathbf{x}, \tilde{\mathbf{x}}) \tag{4.7}$$

Remark 1. *Here, a bounded kernel function is a rather mild condition as most of kernels have finite upper bound, e.g., the absolute value of the cosine kernel is always less than one and the linear kernel is bounded as long as the support* Δ *of* $p(\mathbf{y})$ *is compact.*

We leave the proof of the theorem in Section 4.6.

The empirical target similarity function s_n can be rewritten in a compact matrix form as

$$\begin{aligned} s_n(\mathbf{x}, \tilde{\mathbf{x}}) &= \sum_{i,j=1}^{n} \{T(\mathbf{x}, \mathbf{y}_i) T(\tilde{\mathbf{x}}, \mathbf{y}_j) k(\mathbf{y}_i, \mathbf{y}_j)\} \\ &= \mathbf{v}_T(\mathbf{x})^T K \mathbf{v}_T(\tilde{\mathbf{x}}) \end{aligned} \tag{4.8}$$

where K is an $n \times n$ kernel matrix with $K = [k(\mathbf{y}_i, \mathbf{y}_j)]_{n \times n}$, and the corresponding distance metric d_{tgt} is

$$\begin{aligned} d_{\text{tgt}}(\mathbf{x}, \tilde{\mathbf{x}}) &= s_n(\mathbf{x}, \mathbf{x}) + s_n(\tilde{\mathbf{x}}, \tilde{\mathbf{x}}) - 2 s_n(\mathbf{x}, \tilde{\mathbf{x}}) \\ &= (\mathbf{v}_T(\mathbf{x}) - \mathbf{v}_T(\tilde{\mathbf{x}}))^T K (\mathbf{v}_T(\mathbf{x}) - \mathbf{v}_T(\tilde{\mathbf{x}})) \end{aligned} \tag{4.9}$$

where $\mathbf{v}_T(\cdot)$ defines a mapping

$$\mathbf{v}_T : \mathbb{R}^t \to \mathbb{R}^n, \mathbf{x} \mapsto \mathbf{v}_T(\mathbf{x}) \tag{4.10}$$

from the target space $\mathbb{R}^t$ to an n-dimensional vector space $\mathbb{R}^n$:

$$\mathbf{v}_T(\mathbf{x}) = \begin{bmatrix} T(\mathbf{x}, \mathbf{y}_1) & T(\mathbf{x}, \mathbf{y}_2) & \cdots & T(\mathbf{x}, \mathbf{y}_n) \end{bmatrix}^T \tag{4.11}$$

These n source instances $\mathbf{y}_i, 1 \leq i \leq n$ can be seen as "landmark" instances in the source space, and this mapping summarizes the relevance of the target instance $\mathbf{x}$ to these landmark instances. It asymptotically captures the target metric structure as $n \to +\infty$ by Theorem 3. Note that for ease of notation we discard the constant factor $\frac{1}{n^2}$ in Eq. (4.5) here.

4.3 Transfer Learning of Distance Functions

The transfer function $T(\cdot, \cdot)$ plays the central role in connecting the metric structures in target and source spaces as shown in Eqs. (4.5) and (4.9). To learn the transfer function, two aspects can be explored to reveal the intrinsic distance structure in the target space.

The most direct component which provides the connection between the source and target domains is a set $\mathcal{C} = \{(\mathbf{x}_k, \mathbf{y}_k)\}$ of observed pairs of relevant instances between the two domains. For example, this can be images and their surrounding text; or the equivalent English translation to a Chinese document. This provides the bridge needed for transfer learning of metrics across heterogeneous spaces.

In the cross-domain metric sampling process, only source similarity is sampled to compute the target similarity. On the other hand, a priori information about the structure of the target distance is directly available in the *original target* space. We refer to this as *structural information* about the target space. The learned distance should inherit the metric structures of the original target space as well. Specifically, given a set of target instances, let $Q_{p,q}$ denote the similarity between two instance $\mathbf{x}_p$ and $\mathbf{x}_q$, $1 \leq p, q \leq m$ in the original target space. Then they can be utilized to make the target distance Eq. (4.9) consistent with the metric structure of the original target space. Moreover, aligning source and target metric structures also maximizes the cross-domain correlations, which equivalently imposes a global consistency prior to link the relevant instances in heterogeneous domains. We will reveal this connection in the later.

Now we propose an algorithm in order to optimize the distance transfer process between the two spaces. The optimization problem over the transfer

function T is defined as follows:

$$\min_{T} \gamma \mathcal{L}_{\varepsilon}(T, \mathcal{C}) + \frac{\eta}{2} \sum_{p,q=1}^{m} g\left(Q_{p,q}, d_{\mathrm{tgt}}\left(\mathbf{x}_p, \mathbf{x}_q\right)\right) + \Omega\left(T\right) \tag{4.12}$$

The expression in Eq. (4.12) measures the effectiveness of the distance transfer process, with the corresponding balancing parameters γ and η.

- The first term encodes how the source and target spaces are linked by T in $\mathcal{C}$. As aforementioned, the transfer function T measures the probability of source and target instances being relevant to each other. Based on this probabilistic explanation, we choose the negative logistic loss to estimate the transfer function by maximizing the likelihood over the pairs of the relevant instances in $\mathcal{C}$:

$$\begin{aligned} &\mathcal{L}_{\varepsilon}(T, \mathcal{C}) \\ &= \textstyle\sum_{\mathcal{C}} - \log\left\{(1-\varepsilon)T(\mathbf{x}_k, \mathbf{y}_k) + \varepsilon(1 - T(\mathbf{x}_k, \mathbf{y}_k))\right\} \end{aligned} \tag{4.13}$$

  Here we consider the noise in $\mathcal{C}$, which flips a pair of irrelevant source and target instances to a relevant one in $\mathcal{C}$ with probability $\varepsilon \in [0, 1]$. By minimizing the objective function in Eq. (4.12) alternately between ε and T in a coordinate descent manner, they can be simultaneously inferred. When fixing T, minimizing w.r.t. ε is a standard convex optimization problem. In Section 4.3.2, we will present the optimization of T with fixed ε. Minimizing this term makes the output of the transfer learning process consistent with observations of the paired source and target samples, so that the transfer function has larger output on a pair of target and source instances in $\mathcal{C}$.

- The second term measures the consistency of the target distance with the structural information about the original target metric space. We choose the loss function $g\left(Q_{p,q}, d_{\mathrm{tgt}}\left(\mathbf{x}_p, \mathbf{x}_q\right)\right) = Q_{p,q} d_{\mathrm{tgt}}\left(\mathbf{x}_p, \mathbf{x}_q\right)$ in this chapter. If two target instances are similar according to $Q_{p,q}$, their target distances are minimized; otherwise, their distances will be maximized.

- The last term $\Omega\left(T\right)$ regularizes learning of the transfer learning process, which will be extended in Section 4.3.1 when establishing the transfer function.

We note that the expression in Eq. (4.12) contains several terms, the most important of which correspond to the effects of the co-occurrence data and auxiliary data in the effectiveness of the distance function. The relative importance of co-occurrence data and auxiliary data in the objective function are regulated by the balancing parameters γ and η. The expression discussed above is an optimization problem designed to determine the best translator function T. However, in order to determine this optimum function, we need to further express it is the form of other simplified semantic topic space matrices. This results in a closed-form description of the translator function, whose parameters can be optimized. The decomposition of T into semantic topic spaces will be discussed in the next section.

4.3.1 Designing the Transfer Function

The source and target spaces are quite different in terms of their feature representation. To establish their connection, we must discover a common structure which can link them together. It is possible to discover some common factors to describe the heterogeneous instances simultaneously. For example, a text document usually contains several topics which describe different aspects of the underlying concepts at a higher level. In a web page depicting *bird*, the related topics, such as the head, body and tail, are described in its textual part. Meanwhile, there is a corresponding *bird* image illustrating them. By aligning the topics of the text (i.e., the source instances) and images (i.e., the target instances) in a space with several unspecified topics, they can be semantically linked together by investigating their co-occurrence data. For this purpose, we construct two transformation matrices U and V to map the source and target instances into a common space with r unspecified factors to link heterogeneous domains as follows. This dimensionality is essentially the number of topics, because each dimension in this space represents a latent topic for semantic correspondence. We will show that the translator function can be expressed in terms of these topic spaces, and therefore the key to finding an optimal translator function T is to determine the optimal translation matrices U and V. The matrices U and

V are defined as follows.

$$\begin{aligned} U &\in \mathbb{R}^{r\times s} : \mathbb{R}^s \to \mathbb{R}^r, \mathbf{y} \mapsto U\mathbf{y}, \\ V &\in \mathbb{R}^{r\times t} : \mathbb{R}^t \to \mathbb{R}^r, \mathbf{x} \mapsto V\mathbf{x} \end{aligned} \tag{4.14}$$

Then, the transfer function T is a function of the source and target instances as

$$T(\mathbf{x}, \mathbf{y}) = f(\langle V\mathbf{x}, U\mathbf{y}\rangle) = f\left(\mathbf{x}^T V^T U\mathbf{y}\right) = f\left(\mathbf{x}^T S\mathbf{y}\right) \tag{4.15}$$

where $\langle\cdot,\cdot\rangle$ denotes the inner product, and the matrix S is used to briefly denote $V^T U$; f is the activation function acting on $\mathbf{x}^T S\mathbf{y}$. We choose the logistic sigmoid function as f, i.e., $f(\theta) = \dfrac{1}{1+e^{-\theta}}$. It is differentiable and real-valued in the interval $[0, 1]$. In this case, $T(\mathbf{x}, \mathbf{y})$ outputs the probability that $\mathbf{x}$ and $\mathbf{y}$ are a pair of the relevant target and source instances.

We can use the conventional squared norm $\Omega(T) = \dfrac{1}{2}\left(\|U\|_F^2 + \|V\|_F^2\right)$ to regularize the transfer function T on two transformations respectively, where $\|\cdot\|_F$ is the Frobenius norm. However, since this $\Omega(T)$ is not convex, the global minima cannot be guaranteed by a solution. Fortunately, it is possible to learn S directly by the trace norm as in [63], [64]. It is defined as follows

$$\|S\|_\Sigma = \inf_{S=U^T V} \frac{1}{2}\left(\|U\|_F^2 + \|V\|_F^2\right) \tag{4.16}$$

The trace norm is a convex function of S, and can be computed as the sum of its singular values. It is a surrogate of matrix rank [65], and minimizing it can limit the dimensionality r of the latent factor space. In other words, *minimizing the trace norm results in the fewest topics to explain the correspondence between text and images.* This regularizes the transfer function by the preference to a small size of intermediate topics to link heterogeneous domains as stated in the information bottleneck method [66].

4.3.2 Implementation Details

We use the second term in Eq. (4.12) to leverage the similarity structure in original target space. The loss function penalizes the large distance between

similar instances. We can rewrite this term as

$$\begin{aligned}
&\frac{1}{2}\sum_{p,q=1}^{m} g(Q_{p,q}, d_{\mathrm{tgt}}(\mathbf{x}_p, \mathbf{x}_q)) = \frac{1}{2}\sum_{p,q=1}^{m} Q_{p,q} d_{\mathrm{tgt}}\left(\mathbf{x}_p, \mathbf{x}_q\right) \\
&= \sum_{p,q=1}^{m} Q_{p,q} \mathbf{v}_T\left(\mathbf{x}_p\right)^T K \mathbf{v}_T\left(\mathbf{x}_p\right) \\
&- \sum_{p,q=1}^{m} Q_{p,q} \mathbf{v}_T\left(\mathbf{x}_p\right)^T K \mathbf{v}_T\left(\mathbf{x}_q\right) \\
&= \mathrm{tr}\left(\Xi(S)^T K \Xi(S) D\right) - \mathrm{tr}\left(\Xi(S)^T K \Xi(S) Q\right) \\
&= \mathrm{tr}\left(K \Xi(S) L \Xi(S)^T\right)
\end{aligned} \tag{4.17}$$

where $\Xi(S) = \left[\mathbf{v}_T\left(\mathbf{x}_1\right), \mathbf{v}_T\left(\mathbf{x}_2\right), \cdots, \mathbf{v}_T\left(\mathbf{x}_m\right)\right]$ is an $n \times m$ matrix dependent on S, and tr denotes the trace operation of a matrix. D is a diagonal $m \times m$ matrix with each diagonal element being the corresponding row summation of Q, and $L = D - Q$ is the Laplacian matrix. Then the objective function in Eq. (4.12) with fixed ε can be rewritten as

$$\begin{aligned}
&\min_S \gamma \sum_{\mathcal{C}} -\log\left\{(1-\varepsilon) f(\mathbf{x}_k^T S \mathbf{y}_k) + \varepsilon(1 - f(\mathbf{x}_k^T S \mathbf{y}_k))\right\} \\
&+\eta \mathrm{tr}\left(K \Xi(S) L \Xi(S)^T\right) + \|S\|_{\Sigma}
\end{aligned} \tag{4.18}$$

The objective function of Eq. (4.18) contains non-differentiable trace norm regularizer and a differentiable part. In order to represent the objective function of Eq. (4.18) more succinctly, we introduce the differentiable part $F\left(S\right)$ as

$$\begin{aligned}
&F\left(S\right) \\
&= \gamma \sum_{\mathcal{C}} -\log\left\{(1-\varepsilon) f(\mathbf{x}_k^T S \mathbf{y}_k) + \varepsilon(1 - f(\mathbf{x}_k^T S \mathbf{y}_k))\right\} \\
&+\eta \mathrm{trace}\left(K \Xi(S) L \Xi(S)^T\right)
\end{aligned} \tag{4.19}$$

Then, the objective function of Eq. (4.19) can be rewritten as $F\left(S\right) + \|S\|_{\Sigma}$. For the differentiable part $F(S)$, its gradient $\nabla F\left(S\right)$ can be computed as

$$\begin{aligned}
&\nabla F\left(S\right) = \gamma \sum_{\mathcal{C}} \left\{ -\frac{(1-2\varepsilon) f'(a_k)}{(1-\varepsilon) f(a_k) + \varepsilon(1 - f(a_k))} \mathbf{x}_k \mathbf{y}_k^T \right\} \\
&+\eta \Gamma
\end{aligned} \tag{4.20}$$

where f' is the derivative of f, $a_k = \mathbf{x}_k^T S \mathbf{y}_k$, and Γ is the $t \times s$ gradient matrix

of $\mathrm{tr}\left(K\Xi(S)L\Xi(S)^T\right)$ w.r.t. S, whose (u,v)th element can be computed as

$$\Gamma_{uv} = \frac{\partial \mathrm{tr}\left(K\Xi(S)L\Xi(S)^T\right)}{\partial S_{uv}} = 2\mathrm{tr}\left[\left(K\Xi(S)L\right)^T \frac{\partial \Xi(S)}{\partial S_{uv}}\right] \tag{4.21}$$

Here $\dfrac{\partial \Xi(S)}{\partial S_{uv}}$ is an $n \times m$ matrix, and its (i,j)th element is

$$\left[\frac{\partial \Xi(S)}{\partial S_{uv}}\right]_{ij} = f'(\mathbf{x}_j^T S \mathbf{y}_i) X_{ju} Y_{iv}, \tag{4.22}$$

Denote $X = [\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_m]^T$ and $Y = [\mathbf{y}_1, \mathbf{y}_2, \cdots, \mathbf{y}_n]^T$ are $m \times t$ and $n \times s$ data matrices, then X_{ju} and Y_{iv} are the uth and vth dimensional features in $\mathbf{x}_j$ and $\mathbf{y}_i$, respectively. Combining Eqs. (4.21) and (4.22), with some algebraic operations, the gradient matrix Γ can be rewritten in a compact form as

$$\Gamma = X^T \left(K\Xi(S)L \circ H\right)^T Y \tag{4.23}$$

where $\circ$ denotes the element-wise product of two matrices, and H is an $n \times m$ matrix with its elements as $H_{ij} = f'(\mathbf{x}_j^T S \mathbf{y}_i)$.

We apply the proximal gradient method [46] to minimize the loss function with trace norm regularizer. In order to optimize this objective function, the proximal gradient method quadratically approximates it by Taylor expansion at current S_τ and Lipschitz coefficient α as follows

$$\begin{aligned} Q\left(S, S_\tau\right) = &\frac{\alpha}{2}\left\|S - G_\tau\right\|_F^2 + \left\|S\right\|_\Sigma + F\left(S_\tau\right) \\ &-\frac{1}{2\alpha}\left\|\nabla F\left(S_\tau\right)\right\|_F^2 \end{aligned} \tag{4.24}$$

and

$$G_\tau = S_\tau - \alpha^{-1}\nabla F\left(S_\tau\right) \tag{4.25}$$

Algorithm 3 summarizes the proximal gradient based method to optimize the expression in Eq. (4.18). As shown, S can be updated by minimizing $Q\left(S, S_\tau\right)$ with the fixed S_τ iteratively. This can be solved by singular value thresholding [65] in line 4 in Algorithm 3. As pointed out in [46], the convergence of the proximal gradient algorithm in loop 2-5 can be accelerated by making an initial estimate of α and increasing it by a constant factor λ until $F\left(S_{\tau+1}\right) + \left\|S_{\tau+1}\right\|_\Sigma \leq Q\left(S_{\tau+1}, S_\tau\right)$.

Algorithm 3 Proximal Gradient Solver for (4.18) with Fixed ε

input Correspondence set $\mathcal{C}$, source kernel matrix K, and Laplacian matrix L, balancing parameters γ and η.

1 Initialize $S_\tau \leftarrow 0$ and $\tau \leftarrow 0$.
repeat
repeat
2 Initialize $\alpha \leftarrow \alpha_0$.
3 Set $G_\tau = S_\tau - \alpha^{-1}\nabla F(S_\tau)$.
4 Update $S_{\tau+1} \leftarrow U\mathrm{diag}\left(\sigma - \frac{\gamma}{\alpha}\right)_+ \mathrm{V}^{\mathrm{T}}$. Here $U\mathrm{diag}(\sigma)\mathrm{V}^{\mathrm{T}}$ gives the SVD of G_τ.
5 Set $\alpha \leftarrow \lambda\alpha$
until $F(S_{\tau+1}) + \|S_{\tau+1}\|_\Sigma \leq Q(S_{\tau+1}, S_\tau)$.
6 $\tau \leftarrow \tau + 1$.
until Convergence or maximum iteration number achieves.

4.4 Related Work

Various methods have been proposed to learn distance metric by leveraging the correspondence knowledge across heterogeneous domains [50], [54]. In [50], *multi-label distance metric learning* (ML-DML) is proposed to learn a distance metric on the target space from the observed occurrence between source and target instances. It explores the semantic correlation of images and the keywords in the associated text documents, and learns a Mahalonobis metric in closed form. The problem of learning the distance metric in target spaces can also be seen as a kind of transfer learning from heterogeneous data in different feature spaces. The work in [32] proposes *heterogeneous transfer learning* (HTL) algorithm, which uses both text and visual words as source information to extract a new latent feature representation for each image, which could be used to compute a new distance metric in the target image space. However, both of these algorithms do not explore the problem of transfer learning of distance metrics. As already mentioned, we assume the metric structure has a smaller semantic gap between the low-level features and high-level semantic concepts in the source space. The goal of this chapter is to transfer this metric structure into the target space, which can result in more effective distance functions in the target space. To the best of our knowledge, the method in this chapter is one of the first to demonstrate how to "translate" distance structures across heterogeneous domains and show the results for the case of a practical problem.

Finally, we distinguish the proposed translator function from other latent models. Previous latent methods, such as latent semantic analysis [26], probabilistic latent semantic analysis [27] and latent Dirichlet allocation [28], are restricted to latent factor discovery from the co-occurrence observations. On the contrary, in this chapter, the goal of our approach is to establish the correspondence between the underlying distance metrics in the source and target space so that in the target space the obtained target feature space has a tractable semantic gap. To the best of our knowledge, it is one of the first algorithms to address such a heterogeneous distance transfer problem.

4.5 Experiments

In this section, we compare the proposed distance metrics derived from the transfer learning process to other natural distance metrics which are typically used for a variety of applications. We will show that our approach provides superior results to the other methods.

One challenge is to design a method for qualitative evaluation of the distance metrics. Since distance metrics are inherently semantic functions which are used as subroutines in the context of different kinds of applications, it is natural to test the effectiveness of using different kinds of distance functions on a particular application in order to measure its quality. For example, one can test the effectiveness of a nearest neighbor classifier with the use of different kinds of distance metrics. The idea is that a distance function which retains the most meaningful aspects of the feature space, and adjusts for the most noisy aspects is most likely to work effectively within the context of an application such as classification. In general, for unsupervised problems such as clustering and distance function design, qualitative tests on real data are generally intended to be designed in an evidentiary way, so as to provide an understanding of the advantages of using a particular kind of approach for distance function design.

4.5.1 Data Sets

In order to test our approach we needed paired image and text documents. Furthermore, since we used classification as our base application, we also

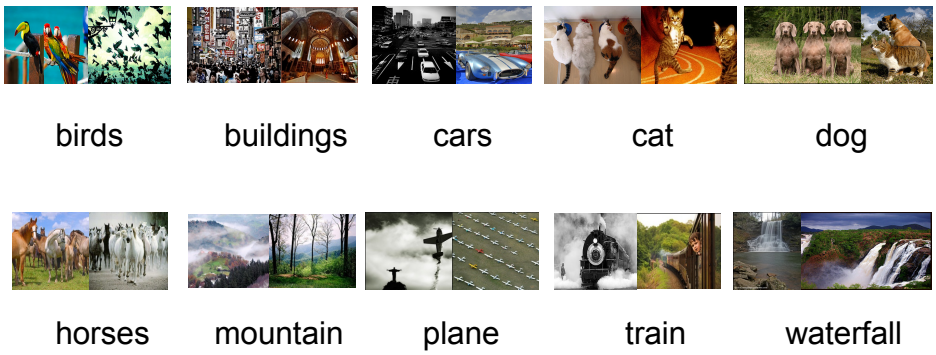


Figure 4.2: Illustration of example images in the data set.

Table 4.1: The number of the crawled web pages by each query. By using the category names as query keywords, the returned web pages are crawled. The images in these web pages are also collected.

Category	Crawled web pages	Category	Crawled web pages
birds	930	horses	654
buildings	9216	mountain	4153
cars	728	plane	1356
cat	229	train	457
dog	486	waterfall	22006

Table 4.2: The number of images in each category for performance evaluation. For performance evaluation, all the images are manually annotated with ground truth by human annotators for evaluation purpose.

Category	Number of positive examples	Number of negative examples	Category	Number of positive examples	Number of negative examples
birds	338	349	horses	263	268
buildings	2301	2388	mountain	927	1065
cars	120	125	plane	509	549
cat	67	72	train	52	53
dog	132	142	waterfall	5153	5737

needed some class labels on the images in order to test the effectiveness of the distance function learning process. The data sets consist of the Corel image data set and a collection of Flickr web pages. Figure 4.2 illustrate some examples of images in the data set. We use 10 categories to evaluate the effectiveness on the image classification task. To collect paired image and

text collections for experiments, the names of these 10 categories are used as query keywords to crawl web pages from the Flickr web site and Wikipedia. Table 4.1 shows the number of crawled web pages for each category. Flickr is an image sharing web site, where the users can share images with their friends and other users, and make textual comments and tags on the shared images. In each crawled web page, the images and the corresponding text documents are used to establish correspondence between text and images. The textual parts of the crawled web pages are used as source instances for metric transfer, and the images are used as auxiliary images in the training set.

For images, visual features are extracted in order to construct a multi-dimensional representation. These include 500 dimensional bag-of-word feature representation quantized from SIFT descriptor. χ^2 similarity between the target instances is used as $Q_{p,q}$ to provide metric information in the original target space. For text documents, all the tokens are extracted and stemmed, and the remaining term frequencies are used as textual features in experiments. For each category, the images are manually annotated by human annotators to collect the ground truth labels for evaluation purpose as shown in Table 4.2. Nearly the same number of images are collected as the negative examples. These images contain the objects of the different categories. These categories are not exclusive which means one image can be annotated with more than one category. Accordingly the following experiments are conducted in such a multi-label case with binary labels for each category.

4.5.2 Compared Algorithms

We use the following algorithms and baselines in order to test the effectiveness of our distance-transfer process.

- As the baseline, we directly compute the Euclidean distance between images based on their visual features. We refer to this metric as ED. This method does not use any of the additional information available in corresponding text in order to improve the quality of the distance function.

- The Kernel Multi-Label Distance Metric Learning [50] algorithm com-

putes the image distance from the co-occurrence between image and text instances. We refer to this algorithm as KML-DML. The Gaussian kernel on the image domain is used here.

- The Heterogeneous Transfer Learning [54] method is a classification algorithm across heterogeneous spaces. Relational matrix between images and text documents is factorized to extract the implicit representation of target instances, based on which the distance function can be set up. We refer to this method as HTL.

- Finally, we test the proposed method in this chapter with two kinds of text similarity measures k. One uses the linear similarity of inner product of text vectors and the other is the typical cosine similarity between text vectors. They are two of the most effective kernel similarities used for text corpus. We denote the distance translators associated with these two text similarity measures by "DT-Lin" and "DT-Cos", respectively. We refer to this method as DT, with specific instantiations as DT-Lin and DT-Cos respectively.

The nearest neighbor (NN) classifier is applied to classify the images based on the above learned distances to compare their performance in classifying the images. For each image category, ten positive examples and ten negative images are randomly selected as labeled instances for the classifiers, and the remaining are used for testing. This process is repeated five times. The error rate and the associated standard deviation for each category is reported. We also use a varying number of text documents as landmark source instances to construct the distance, and compare the corresponding results with related algorithms. All the parameters are tuned based on a twofold cross-validation procedure on the selected training set, and the parameters with the best performance are selected to train the models.

4.5.3 Results

Next, we present the error rates of the classifiers with the use of this nearest neighbor metric. Table 4.3 compares different algorithms in terms of their classification error rates. In this case, we used $2,000$ associated images and text documents in order to learn the distance metric in the image space.

Table 4.3: Comparison of error rates and the deviations of the proposed distance transfer algorithms (DT-Lin and DT-Cos) compared with the other state-of-the-art transfer methods over ten categories. Our results in bold achieve smaller error rates than the other existing algorithms.

Category	ED	KML-DML	HTL
birds	0.2639±0.0012	0.2481±0.0008	0.2619±0.0015
buildings	0.2856±0.0002	0.2625±0.0004	0.2707±0.0021
cars	0.3027±0.0073	0.2414±0.0054	0.3065±0.0030
cat	0.2755±0.0043	0.3333±0.0040	0.2525±0.0038
dog	0.2252±0.0039	0.1802±0.0057	0.2343±0.0037
horses	0.2667±0.0019	0.3000±0.0015	**0.2500±0.0021**
mountain	0.3176±0.0010	0.2974±0.0008	0.3097±0.0003
plane	0.2667±0.0009	0.2633±0.0011	0.2133±0.0008
train	0.2716±0.0029	0.2593±0.0068	0.2716±0.0118
waterfall	0.2611±0.0008	0.2476±0.0015	0.2435±0.0009

Category	DT-Lin	DT-Cos
birds	**0.2421±0.0010**	0.2559±0.0011
buildings	**0.2157±0.0000**	**0.2145±0.0004**
cars	**0.2107±0.0044**	**0.2031±0.0026**
cat	0.3131±0.0084	0.2929±0.0053
dog	**0.1802±0.0027**	**0.1712±0.0031**
horses	0.2517±0.0014	**0.2467±0.0018**
mountain	**0.2974±0.0005**	**0.2952±0.0005**
plane	0.2633±0.0009	0.2617±0.0005
train	**0.1924±0.0058**	**0.1852±0.0049**
waterfall	**0.2409±0.0002**	**0.2425±0.0001**

We will also compare the average error rates by using different numbers of text documents. From this result, we can find that among all ten categories, the proposed distance transfer, both DT-Lin and DT-Cos, performs the best on seven categories as compared with the existing methods, respectively. Moreover, as illustrated in Figure 4.3, in terms of average rates, both DT-Lin and DT-Cos gain a significant improvement compared with other algorithms.

As stated in Section 4.2, the text documents play an important landmark role of embedding the image instances by the distance transfer learning process. With more landmark source instances, the empirical target metric asymptotically converges to the true one. Therefore, it is instructive to examine the effect of increasing the number of such landmarks. In Figure 4.3, we illustrate the effectiveness of different algorithms with varying number of text documents. The number of documents is illustrated on the X-axis, whereas the error rate is illustrated on the Y-axis. As we can see, the error rates of the DT-Lin and DT-Cos algorithms are reduced with an increasing number of documents in the source space since more information about source metric structure is transferred to the target space. We also note that their improvements are more significant than other algorithms when more text documents are involved. This suggests that there is a real gain in the quality of the distance function through the process of transfer learning from text to images.

4.5.4 Computing Time

Finally, we compare the computational efficiency of the different algorithms for learning the target distance metric. All the algorithms are conducted on the same computing platform with 2.10 GHz Intel CPU and 3 GB physical memory. Since the Euclidean metric is directly available without any learning process, we omit its computing time here. Table 4.4 shows the computing time with $2,000$ text documents for learning the distance. DT-Lin and DT-Cos are much faster than HTL but slower than KML-DML, since KML-DML has a closed-form solution when learning the metric and involves only one matrix inversion operation [50].

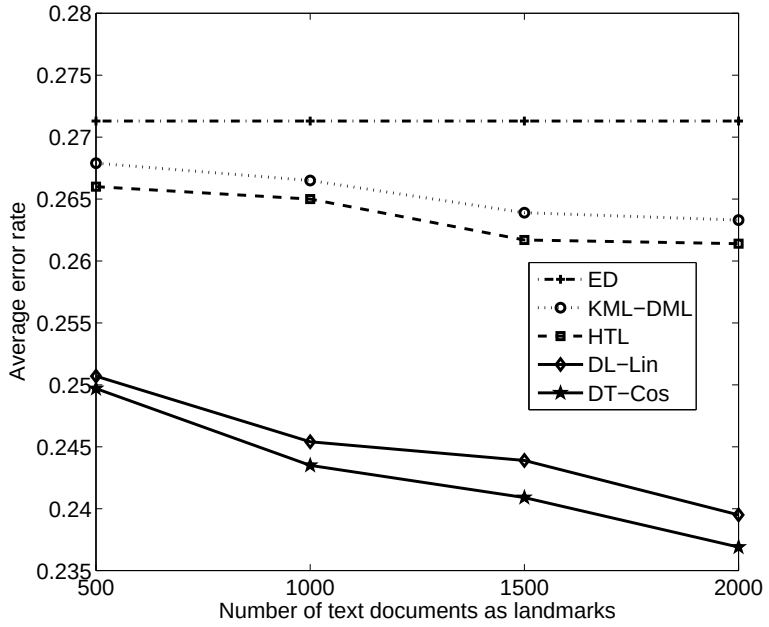


Figure 4.3: Average error rates of compared algorithms with varying number of text documents as landmark source instances.

Table 4.4: Comparison of computing time (in seconds) of different algorithms for learning the target distance metric.

Category	Computing Time
ED	N/A
KML-DML	562.52
HTL	4536.07
DT-Lin	678.93
DT-Cos	719.25

4.6 Proof of Convergence

Proof of Theorem 2

Here we prove the Theorem 2. We first prove the following two lemmas.

Lemma 1.

$$\mathbb{E}s_n\left(\mathbf{x},\tilde{\mathbf{x}}\right)=s\left(\mathbf{x},\tilde{\mathbf{x}}\right)+\frac{1}{n}\varrho\left(\mathbf{x},\tilde{\mathbf{x}}\right)$$

where

$$\varrho\left(\mathbf{x},\tilde{\mathbf{x}}\right)=\mathbb{E}_{\mathbf{y}}\left[T\left(\mathbf{x},\mathbf{y}\right)T\left(\tilde{\mathbf{x}},\mathbf{y}\right)k\left(\mathbf{y},\mathbf{y}\right)\right]-s\left(\mathbf{x},\tilde{\mathbf{x}}\right)$$

Proof.

$$\begin{aligned}
&\mathbb{E}s_n(\mathbf{x},\tilde{\mathbf{x}}) = \mathbb{E}\frac{1}{n^2}\sum_{i,j=1}^{n} T(\mathbf{x},\mathbf{y}_i)\,T(\mathbf{x},\mathbf{y}_j)\,k(\mathbf{y}_i,\mathbf{y}_j)\\
&= \frac{1}{n^2}\sum_{i,j=1}^{n}\mathbb{E}\left[T(\mathbf{x},\mathbf{y}_i)\,T(\mathbf{x},\mathbf{y}_j)\,k(\mathbf{y}_i,\mathbf{y}_j)\right]\\
&= \frac{1}{n^2}\sum_{i\neq j,i,j=1}^{n}\mathbb{E}\left[T(\mathbf{x},\mathbf{y}_i)\,T(\mathbf{x},\mathbf{y}_j)\,k(\mathbf{y}_i,\mathbf{y}_j)\right]\\
&+\frac{1}{n^2}\sum_{i=1}^{n}\mathbb{E}_{\mathbf{y}_i}\left[T(\mathbf{x},\mathbf{y}_i)\,T(\mathbf{x},\mathbf{y}_i)\,k(\mathbf{y}_i,\mathbf{y}_i)\right]\\
&= \frac{n(n-1)}{n^2}s(\mathbf{x},\tilde{\mathbf{x}}) + \frac{1}{n^2}\sum_{i=1}^{n}\mathbb{E}_{\mathbf{y}}\left[T(\mathbf{x},\mathbf{y})\,T(\mathbf{x},\mathbf{y})\,k(\mathbf{y},\mathbf{y})\right]\\
&= s(\mathbf{x},\tilde{\mathbf{x}}) + \frac{1}{n}\left\{\mathbb{E}_{\mathbf{y}}\left[T(\mathbf{x},\mathbf{y})\,T(\mathbf{x},\mathbf{y})\,k(\mathbf{y},\mathbf{y})\right] - s(\mathbf{x},\tilde{\mathbf{x}})\right\}\\
&= s(\mathbf{x},\tilde{\mathbf{x}}) + \frac{1}{n}\varrho(\mathbf{x},\tilde{\mathbf{x}})
\end{aligned}$$

□

This lemma shows that $\mathbb{E}s_n(\mathbf{x},\tilde{\mathbf{x}}) \to s(\mathbf{x},\tilde{\mathbf{x}})$ as $n \to +\infty$.

Lemma 2. *Let $s_n^{i,\mathbf{z}}(\mathbf{x},\tilde{\mathbf{x}})$ be the empirical estimator of s with the ith source instance $\mathbf{y}_i$ replaced with $\mathbf{z}$. Then we have*

$$\left|s_n^{i,\mathbf{z}}(\mathbf{x},\tilde{\mathbf{x}}) - s_n(\mathbf{x},\tilde{\mathbf{x}})\right| \leq \frac{2B}{n}$$

where B is the upper bound of the kernel function, i.e., $|k(\mathbf{y},\mathbf{z}) < B|$ for any $\mathbf{y},\mathbf{z}$.

Proof.

$$\begin{aligned}
&|s_n^{i,\mathbf{z}}(\mathbf{x},\tilde{\mathbf{x}}) - s_n(\mathbf{x},\tilde{\mathbf{x}})|\\
&= |\frac{1}{n^2}\sum_{j=1}^{n} T(\mathbf{x},\mathbf{z})\,T(\mathbf{x},\mathbf{y}_j)\,k(\mathbf{z},\mathbf{y}_j)\\
&-\frac{1}{n^2}\sum_{j=1}^{n} T(\mathbf{x},\mathbf{y}_i)\,T(\mathbf{x},\mathbf{y}_j)\,k(\mathbf{y}_i,\mathbf{y}_j)|\\
&= \frac{1}{n^2}|\sum_{j=1}^{n} T(\mathbf{x},\mathbf{y}_j)\left\{T(\mathbf{x},\mathbf{z})\,k(\mathbf{z},\mathbf{y}_j)\right.\\
&\left.-T(\mathbf{x},\mathbf{y}_i)\,k(\mathbf{y}_i,\mathbf{y}_j)\right\}|\\
&\leq \frac{1}{n^2}\sum_{j=1}^{n}|T(\mathbf{x},\mathbf{y}_j)|\left\{|T(\mathbf{x},\mathbf{z})\,k(\mathbf{z},\mathbf{y}_j)|\right.\\
&\left.+|T(\mathbf{x},\mathbf{y}_i)\,k(\mathbf{y}_i,\mathbf{y}_j)|\right\}\\
&\leq \frac{1}{n^2}\sum_{j=1}^{n}\left\{|k(\mathbf{z},\mathbf{y}_j)| + |k(\mathbf{y}_i,\mathbf{y}_j)|\right\}\\
&\leq \frac{1}{n^2}\cdot 2nB = \frac{2B}{n}
\end{aligned}$$

The second inequality applies the fact that $T(\mathbf{x}, \mathbf{y}) \leq 1$. $\square$

Now we revisit McDiarmid inequality [61] here.

Theorem 4. *(McDiarmid Inequality) Given random variables* $\{\mathbf{y}_i, 1 \leq i \leq n\}$*, z, and a function* $F(\mathbf{y}_1, \mathbf{y}_2, \cdots, \mathbf{y}_n)$ *which satisfies*

$$\begin{aligned} &\sup_{\mathbf{y}_1, \mathbf{y}_2, \cdots, \mathbf{y}_n, \mathbf{z}} |F(\mathbf{y}_1, \mathbf{y}_2, \cdots, \mathbf{y}_n) \\ &-F(\mathbf{y}_1, \mathbf{y}_2, \cdots, \mathbf{y}_{i-1}, \mathbf{z}, \mathbf{y}_{i+1} \cdots, \mathbf{y}_n)| \leq c_i \end{aligned}$$

then the following inequality holds

$$\begin{aligned} &p\left(|F(\mathbf{y}_1, \mathbf{y}_2, \cdots, \mathbf{y}_n) - \mathbb{E}F(\mathbf{y}_1, \mathbf{y}_2, \cdots, \mathbf{y}_n)| > \varepsilon\right) \\ &\leq 2\exp\left(-\frac{2\varepsilon^2}{\sum_{i=1}^n c_i^2}\right) \end{aligned}$$

Combining Lemma 1 and Lemma 2, applying McDiarmid inequality, we obtain the following theorem

Theorem 5.

$$\begin{aligned} &p\left(\left|s_n\left(\mathbf{x}, \tilde{\mathbf{x}}\right) - \left(s\left(\mathbf{x}, \tilde{\mathbf{x}}\right) + \frac{1}{n}\varrho\left(\mathbf{x}, \tilde{\mathbf{x}}\right)\right)\right| > \varepsilon\right) \\ &\leq 2\exp\left(-\frac{\varepsilon^2 n}{2B^2}\right) \end{aligned}$$

Now we prove the Theorem 2 in the main draft. Let $\mu = 2\exp\left(-\frac{\varepsilon^2 n}{2B^2}\right)$, we have $\varepsilon = B\sqrt{\frac{2}{n}\ln\frac{2}{\mu}}$. Then

$$\begin{aligned} &p\left(\left|s_n\left(\mathbf{x}, \tilde{\mathbf{x}}\right) - \left(s\left(\mathbf{x}, \tilde{\mathbf{x}}\right) + \frac{1}{n}\varrho\left(\mathbf{x}, \tilde{\mathbf{x}}\right)\right)\right| \leq \varepsilon\right) \\ &= 1 - p\left(\left|s_n\left(\mathbf{x}, \tilde{\mathbf{x}}\right) - \left(s\left(\mathbf{x}, \tilde{\mathbf{x}}\right) + \frac{1}{n}\varrho\left(\mathbf{x}, \tilde{\mathbf{x}}\right)\right)\right| > \varepsilon\right) \\ &> 1 - 2\exp\left(-\frac{\varepsilon^2 n}{2B^2}\right) \\ &= 1 - \mu \end{aligned}$$

Thus with probability at least $1-\mu$,

$$\begin{aligned}&|s_n(\mathbf{x},\tilde{\mathbf{x}})-s(\mathbf{x},\tilde{\mathbf{x}})|-\tfrac{1}{n}|\varrho(\mathbf{x},\tilde{\mathbf{x}})|\\&\leq\left|s_n(\mathbf{x},\tilde{\mathbf{x}})-\left(s(\mathbf{x},\tilde{\mathbf{x}})+\frac{1}{n}\varrho(\mathbf{x},\tilde{\mathbf{x}})\right)\right|\leq\varepsilon=B\sqrt{\frac{2}{n}\ln\frac{2}{\mu}}\end{aligned}$$

That is,

$$|s_n(\mathbf{x},\tilde{\mathbf{x}})-s(\mathbf{x},\tilde{\mathbf{x}})|\leq\frac{1}{n}|\varrho(\mathbf{x},\tilde{\mathbf{x}})|+B\sqrt{\frac{2}{n}\ln\frac{2}{\mu}}$$

As $n\to+\infty$, $s_n(\mathbf{x},\tilde{\mathbf{x}})$ will converge in probability at rate $O\left(\frac{1}{\sqrt{n}}\right)$ to $s(\mathbf{x},\tilde{\mathbf{x}})$.

4.7 Conclusion

In this chapter, we propose a transfer learning process for distance metrics, which can effectively transfer the metric information in source domain to learn an effective metric structure in the target domain. For this purpose, as a bridge, we learn the distance transfer by exploring the correspondence information between the source and target spaces. The distance metric in the target space can then be constructed by embedding the target instances into a new feature vector space by a set of landmarks in the source space. The proposed method is compared with existing metric learning algorithms, and the competitive results are achieved.

REFERENCES

[1] B. Zhao, B. I. P. Rubinstein, J. Gemmell, and J. Han, "A Bayesian approach to discovering truth from conflicting sources for data integration," in *Proc. of International Conference on Very Large Databases*, 2012.

[2] J. Pasternack and D. Roth, "Knowing what to believe (when you already know something)," in *Proc. of International Conference on Computational Linguistics*, August 2010.

[3] A. Galland, S. Abiteboul, A. Marian, and P. Senellart, "Corroborating information from disagreeing views," in *Proc. of ACM International Conference on Web Search and Data Mining*, February 2010.

[4] G. Kasneci, J. V. Gael, D. Stern, and T. Graepel, "Cobayes: Bayesian knowledge corroboration with assessors of unknown areas of expertise," in *Proc. of ACM International Conference on Web Search and Data Mining*, 2011.

[5] Y. Bachrach, T. Minka, J. Guiver, and T. Graepel, "How to grade a test without knowing the answers – A Bayesian graphical model for adaptive crowdsourcing and aptitude testing," in *Proc. of International Conference on Machine Learning*, 2012.

[6] X. Yin, J. Han, and P. S. Yu, "Truth discovery with multiple conflicting information providers on the web," in *Proc. of ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, August 2007.

[7] X. L. Dong, L. Berti-Equille, and D. Srivastava, "Integrating conflicting data: The role of source dependence," in *Proc. of International Conference on Very Large Databases*, August 2009.

[8] M. Bilgic, G. Namata, and L. Getoor, "Combining collective classification and link prediction," in *Workshop on Mining Graphs and Complex Structures (at ICDM)*, 2007.

[9] O. Hassanzadeh et al., "A framework for semantic link discovery over relational data," in *CIKM*, 2009.

[10] L. Getoor, N. Friedman, D. Koller, and B. Taskar, "Learning probabilistic models of link structure," *Journal of Machine Learning Research*, no. 3, pp. 679–707, 2002.

[11] M. Girvan and M. Newman, "Community structure in social and biological networks," *Proceedings of the National Academy of Sciences*, vol. 99, no. 12, pp. 7821–7826, June 2002.

[12] A. Clauset, M. E. J. Newman, and C. Moore, "Finding community structure in very large networks," *Physical Review E*, vol. 70, p. 066111, 2004.

[13] Y. Koren, R. Bell, and C. Volinsky, "Matrix factorization techniques for recommender systems," *Computer*, vol. 42, no. 8, pp. 30–37, August 2009.

[14] J. Sethuraman, "A constructive definition of Dirichlet priors," *Statistica Sinica*, vol. 4, pp. 639–650, 1994.

[15] M. Jordan, Z. Ghahramani, T. Jaakkola, and L. Saul, "Introduction to variational methods for graphical models," *Machine Learning*, vol. 37, pp. 183–233, 1999.

[16] X. Yin and W. Tan, "Semi-supervised truth discovery," in *Proc. of International World Wide Web Conference*, March 28–April 1, 2011.

[17] X. Zhou, N. Cui, Z. Li, F. Liang, and T. Huang, "Hierarchical Gaussianization for image classification," Kyoto, Japan, 2009.

[18] M. Gupta, Y. Sun, and J. Han, "Trust analysis with clustering," in *Proc. of International World Wide Web Conference*, April 2011.

[19] K. Kurihara, M. Welling, and N. Vlassis, "Accelerated variational Dirichlet process mixtures," in *NIPS*, 2006.

[20] A. B. Benitez, J. R. Smith, and S.-F. Chang, "Medianet: A multimedia information network for knowledge representation," in *SPIE Proceeding Series*, 2000.

[21] J. Yu, X. Jin, J. Han, and J. Luo, "Social group suggestion from user image collections," in *Proc. of International World Wide Web Conference*, 2010.

[22] S. Sizov, "Geofolk: Latent spatial semantics in web 2.0 social media," in *Proceedings of Third ACM International Conference on Web Search and Data Mining*, 2010.

[23] T. L. Berg, A. C. Berg, and J. Shih, "Automatic attribute discovery and characterization from noisy web images," in *Proc. of European Conference on Computer Vision*, 2010.

[24] S. Wang, Q. Huang, S. Jiang, L. Qin, and Q. Tian, "Visual contextrank for web image re-ranking," in *Proceedings of the First ACM Workshop on Large-Scale Multimedia Retrieval and Mining*, Beijing, China, October 2009, pp. 121–128.

[25] J. Shawe-Taylor and N. Cristianini, *Kernel Methods for Pattern Recognition*. Cambridge University Press, 2004.

[26] T. K. Landauer, P. W. Foltz, and D. Laham, "An introduction to latent semantic analysis," *Discourse Processes*, vol. 25, pp. 259–284, 1998.

[27] T. Hofmann, "Probabilistic latent semantic analysis," in *Uncertainty in Artificial Intelligence*, pp. 289–296, 1999.

[28] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent Dirichlet allocation," *Journal of Machine Learning Research*, vol. 3, pp. 993–1022, 2003.

[29] A. Bosch, A. Zisserman, and X. Munoz, "Scene classification via PLSA," in *Proceedings of the European Conference on Computer Vision*, 2006.

[30] F. Monay and D. Gatica-Perez, "PLSA-based image auto-annotation: Constraining the latent space," in *Proc. of ACM International Conference on Multimedia*, New York, 2004, pp. 348–351.

[31] P. P. Martin Labský, Miroslav Vacura, "Web image classification for information extraction," in *Proceedings of the RAWS 2005 International Workshop on Representation and Analysis of Web Space*, 2005.

[32] Q. Yang, Y. Chen, G. R. Xue, W. Dai, and Y. Yu, "Heterogeneous transfer learning for image clustering via the social web," in *Proceedings of the 47th Annual Meeting of the ACL and the 4th IJCNLP of the AFNLP*, Singapore, August 2009, pp. 1–9.

[33] R. Bekkerman and J. Jeon, "Multi-modal clustering for multimedia collections," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2007.

[34] M. Guillaumin, J. Verbeek, and C. Schmid, "Multimodal semi-supervised learning for image classification," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2010.

[35] A. Rakotomamonjy, F. R. Bach, S. Canu, and Y. Grandvalet, "Simplemkl," *Journal of Machine Learning Research*, vol. 9, pp. 2491–2521, November 2008.

[36] E. P. Xing, A. Y. Ng, M. I. Jordan, and S. Russell, "Distance metric learning, with application to clustering with side-information," in *Proc. of Advanced Neutral Information Processing System*, 2003.

[37] A. W. M. Smeulders, M. Worring, S. Santini, A. Gupta, and R. Jain, "Content-based image retrieval at the end of the early years," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 12, pp. 1349–1380, 2000.

[38] M. Flickner, H. Sawhney, W. Niblack, J. Ashley, Q. Huang, B. Dom, M. Gorkani, J. Hafner, D. Lee, D. Petkovic, D. Steele, and P. Yanker, "Query by image and video content: The QBIC system," in *Intelligent Multimedia Information Retrieval*, Cambridge, MA, USA, 1997, pp. 7–22.

[39] J. R. Bach, C. Fuller, A. Gupta, A. Hampapur, B. Horowitz, R. Humphrey, R. C. Jain, and C.-F. Shu, "Virage image search engine," *Proceedings of SPIE*, vol. 2670, no. 76, 1996.

[40] S. Zhu, K. Yu, Y. Chi, and Y. Gong, "Combining content and link for classification using matrix factorization," in *Proceedings of the 30th ACM SIGIR Conference on Research and Development in Information Retrieval*, 2007.

[41] E. Candés and Y. Plan, "Matrix completion with noise," *Proceedings of the IEEE*, 2009.

[42] E. J. Candés and P. Randall, "Highly robust error correction by convex programming," *IEEE Transactions on Information Theory*, vol. 54, no. 7, pp. 2829–2840, 2006.

[43] J. Wright, A. Ganesh, S. Rao, Y. Peng, and Y. Ma, "Robust principal component analysis: Exact recovery of corrupted low-rank matrices via convex optimization," in *Proceedings of Neural Information Processing Systems*, December 2009.

[44] F. R. K. Chung, "Spectral graph theory," *Regional Conference Series in Mathematics*, vol. 92, 1997.

[45] M. Belkin and P. Niyogi, "Laplacian eigenmaps and spectral techniques for embedding and clustering," in *Proceedings of Advances in Neural Information Processing Systems 14*, Cambridge, MA, USA, 2001, pp. 585–591.

[46] K. C. Toh and S. Yun, "An accelerated proximal gradient algorithm for nuclear norm regularized least squares problems," Preprint on Optimization Online, April 2009.

[47] T.-S. Chua, J. Tang, R. Hong, H. Li, Z. Luo, and Y. Zheng, "Nus-wide: A real-world web image database from National University of Singapore," in *Proc. of ACM International Conference on Image and Video Retrieval*, 2009.

[48] C. G. M. Snoek, M. Worring, J. C. van Gemert, J. M. Geusebroek, and A. W. M. Smeulders, "The challenge problem for automated detection of 101 semantic concepts in multimedia," in *Proceedings of the ACM International Conference on Multimedia*, Santa Barbara, CA, USA, 2006.

[49] J. Tang, S. Yan, R. Hong, G.-J. Qi, and T.-S. Chua, "Inferring semantic concepts from community-contributed images and noisy tags," in *Proc. of ACM International Conference on Multimedia*, 2009.

[50] G.-J. Qi, X.-S. Hua, and H.-J. Zhang, "Learning semantic distance from community-tagged media collection," in *Proc. of International ACM Conference on Multimedia*, 2009.

[51] R. Raina, A. Ng, and D. Koller, "Constructing informative priors using transfer learning," in *Proceedings of Internatinal Conference on Machine Learning*, 2006.

[52] R. Raina, A. Battle, H. Lee, B. Packer, and A. Ng, "Self-taught learning: Transfer learning from unlabeled data," in *Proceedings of Internatinal Conference on Machine Learning*, 2007.

[53] W. Dai, Y. Chen, G.-R. Xue, Q. Yang, and Y. Yu, "Translated learning: Transfer learning across different feature spaces," in *Proceedings of Advances in Neural Information Processing Systems*, 2008.

[54] Y. Zhu, S. J. Pan, Y. Chen, G.-R. Xue, Q. Yang, and Y. Yu, "Heterogeneous transfer learning for image classification," in *Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence*, 2010.

[55] S. V. C. Dorai, "Bridging the semantic gap with computational media aesthetics," *IEEE MultiMedia*, vol. 10, no. 2, pp. 15–17, April 2003.

[56] C. Aggarwal, "Towards systematic design of distance functions for data mining applications," in *Proceedings of the ACM KDD Conference*, 2003.

[57] A. Bar-Hillel, T. Hertz, N. Shental, and D. Weinshall, "Learning distance functions using equivalence relations," in *Proc. of International Conference on Machine Learning*, 2003.

[58] M. Schultz and T. Joachims, "Learning a distance metric from relative comparisons," in *Proc. of Advanced Neutral Information Processing System*, 2004.

[59] K. Q. Weinberger, J. Blitzer, and L. K. Saul, "Distance metric learning for large margin nearest neighbor classification," in *Proc. of NIPS*, 2005.

[60] J. V. Davis, B. Kulis, P. Jain, S. Sra, and I. S. Dhillon, "Information-theoretic metric learning," in *Proc. of International Conference on Machine Learning*, 2007.

[61] O. Bousquet and A. Elisseeff, "Stability and generalization," *Journal of Machine Learning Research*, no. 2, pp. 499–526, March 2002.

[62] R. Jin, S. Wang, and Y. Zhou, "Regularized distance metric learning: Theory and algorithm," in *Proc. of NIPS*, 2009.

[63] N. Srebro, J. Rennie, and T. Jaakkola, "Maximum margin matrix factorization," in *Proceedings of Advances in Neural Information Processing Systems*, 2005.

[64] Y. Amit, M. Fink, N. Srebro, and S. Ullman, "Uncovering shared structures in multiclass classification," in *Proceedings of Internatinal Conference on Machine Learning*, 2007.

[65] J.-F. Cai, E. Candés, and Z. Shen, "A singular value thresholding algorithm for matrix completion," September 2008. [Online]. Available: http://arxiv.org/abs/0810.3286

[66] N. Tishby, F. Pereira, and W. Bialek, "The information bottleneck method," in *Proceedings of the 37th Annual Allerton Conference on Communication, Control and Computing*, pp. 368–377, 1999.